



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## **DETEKCE ANOMÁLIÍ V CHOVÁNÍ DAVU VE VIDEO- DATECH Z DRONU**

CROWD BEHAVIOR ANOMALY DETECTION IN DRONE VIDEODATA

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**BC. DAVID BAŽOUT**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. VÍTĚZSLAV BERAN, Ph.D.**

**BRNO 2021**

## Zadání diplomové práce



Student: **Bažout David, Bc.**  
Program: Informační technologie  
Obor: Počítačové vidění  
Název: **Detekce anomálií v chování davu ve video-datech z dronu**  
**Crowd Behavior Anomaly Detection in Drone Videodata**  
Kategorie: Zpracování obrazu  
Zadání:

1. Seznamte se s metodami počítačového vidění a moderními postupy strojového učení. Zaměřte se na úlohu zpracování obrazových dat (např. z dronu) s cílem detekovat změny chování davu oproti běžnému stavu.
2. Vyberte vhodnou statistickou metodu nebo architekturu CNN a datovou sadu k natrénování vhodného modelu pro analýzu pohybu davu ve snímané scéně z dronu. S modelem experimentujte.
3. Navrhněte systém pro zpracování obrazových dat z dronu, včetně nutných funkcí pro předzpracování dat či jejich výslednou agregaci a prezentaci. Navržený systém realizujte pomocí dostupných knihoven.
4. Vyhodnoťte systém na připravené sadě dat.
5. Prezentujte klíčové vlastnosti řešení formou plakátu a krátkého videa.

### Literatura:

- M. Sonka, V. Hlaváč, R. Boyle. *Image Processing, Analysis, and Machine Vision*, CL-Engineering, ISBN-13: 978-0495082521, 2007.
- Gary R. Bradski, Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*, ISBN 10: 0-596-51613-4, September 2008.
- Dále dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Body 1, 2 a částečně bod 3.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Beran Vítězslav, Ing., Ph.D.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2020

Datum odevzdání: 19. května 2021

Datum schválení: 30. října 2020

## Abstrakt

V posledních letech dochází k rychlému rozvoji využití dronů v mnoha odvětvích. Své uplatnění nachází i v oblasti národních bezpečnostních složek. Cílem této práce je návrh a implementace nástroje provádějícího analýzu davových scén snímaných dronem. Tento nástroj zajišťuje včasnou identifikaci podezřelého chování osob a usnadňuje jeho lokalizaci. Mezi hlavní přínos patří návrh vhodného algoritmu stabilizace videa umožňující stabilizaci drobných záchvěvů, ale i zpětné dohledání ztracené scény. Dále byly navrženy dva detektory anomálie odlišující se ve způsobu extrakce příznakového vektoru a modelování pozadí. Ve srovnání s nejlepšími přístupy současnosti dosahují srovnatelných výsledků, ale zároveň přináší výhodu v podobě možnosti zpracování on-line dat.

## Abstract

There have been lots of new drone applications in recent years. Drones are also often used in the field of national security forces. The aim of this work is to design and implement a tool intended for crowd behavior analysis in drone videodata. This tool ensures identification of suspicious behavior of persons and facilitates its localization. The main benefits include the design of a suitable video stabilization algorithm to stabilize small jitters, as well as trace back of the lost scene. Furthermore, two anomaly detectors were proposed, differing in the method of feature vector extraction and background modeling. Compared to the state of the art approaches, they achieved comparable results, but at the same time they brought the possibility of online data processing.

## Klíčová slova

stabilizace videa, detekce anomálie, optický tok, davové scény, drony

## Keywords

video stabilization, anomaly detection, optical flow, crowded scenes, drones

## Citace

BAŽOUT, Bc. David. *Detekce anomálií v chování davu ve video-datech z dronu*. Brno, 2021. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Vítězslav Beran, Ph.D.

# Detekce anomálií v chování davu ve video-datech z dronu

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Vítězslava Berana Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Bc. David Bažout

15. května 2021

## Poděkování

Chtěl bych využít této příležitosti a poděkovat svému vedoucímu diplomové práce Ing. Vítězslavu Beranovi Ph.D. za dlouhodobou a velmi produktivní spolupráci, která výrazným způsobem prohloubila moje znalosti nejen v oblasti řešení úloh počítačového vidění.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Detekce anomálie v davu</b>	<b>3</b>
2.1	Průzkum metod detekce anomálie . . . . .	3
2.2	Výpočet optického toku . . . . .	7
2.3	Metody stabilizace obrazu . . . . .	9
2.4	Strojové učení . . . . .	15
2.5	Architektury neuronových sítí . . . . .	18
<b>3</b>	<b>Rozbor úlohy a její dekompozice</b>	<b>22</b>
3.1	Rozbor uživatelské úlohy . . . . .	22
3.2	Stabilizace obrazu . . . . .	23
3.3	Systém detekce anomálie . . . . .	25
3.4	Metoda založená na histogramech . . . . .	26
3.5	Metoda založená na extrakci příznaku . . . . .	27
<b>4</b>	<b>Experimenty a vyhodnocení</b>	<b>29</b>
4.1	Datové sady . . . . .	29
4.2	Vyhodnocení metody stabilizace . . . . .	34
4.3	Experimenty s detektory anomálie . . . . .	35
4.4	Vyhodnocení na veřejných datasetech . . . . .	39
<b>5</b>	<b>Uživatelská aplikace</b>	<b>43</b>
5.1	Stabilizátor videa . . . . .	44
5.2	Detektor anomálie . . . . .	45
5.3	Výpočetní modul a klientská aplikace . . . . .	46
<b>6</b>	<b>Závěr</b>	<b>49</b>
	<b>Literatura</b>	<b>50</b>
<b>A</b>	<b>Obsah přiloženého paměťového média</b>	<b>54</b>

# Kapitola 1

## Úvod

Úkolem bezpečnostních složek v rámci společenských událostí s vysokou návštěvností je monitorování mnohdy i rozsáhlých oblastí s omezeným počtem lidských zdrojů. Rychlost zásahu v případě výskytu nebezpečné situace může rozhodovat o záchraně lidských životů. S monitorováním rozsáhlých oblastí již v současnosti pomáhají drony. Dron je po svém vzletu navigován do pozice vhodné pro sledování dané scény, která je ale omezena řadou bezpečnostních předpisů. Zaujímá zde stacionární polohu, nicméně získaná videodata mohou být ovlivněna mnoha nepříznivými vlivy. Uživatel chce být upozorněn na neobvyklé chování davu a potřebuje získat informace o lokalitě jeho výskytu. Při řešení úlohy je nutné vzít v úvahu nedostupnost jakýchkoliv trénovacích dat specifických pro danou scénu.

Cílem této práce je návrh a implementace nástroje umožňujícího automatizované provádění analýzy chování davu a intuitivní lokalizaci oblastí s výskytem neobvyklého chování. V rámci této úlohy je nutné vyřešit nejprve stabilizaci příchozích videodat. Stabilizace videa slouží k odstínění pohybu kamery umístěné na dronu od pohybů ve scéně a je nezbytná i pro následnou lokalizaci v reálné scéně. V druhém kroku probíhá detekce anomálie. Princip detektorů spočívá v modelování obvyklého chování davu pomocí vhodného modelu pozadí. Porovnáním aktuální situace ve scéně s modelem je vypočtena úroveň anomálie [19].

Kapitola 2 vychází z existujících řešení detekce anomálie a její souvislosti s úlohou stabilizace. Hlubší průzkum této problematiky ukázal, že se stále jedná o poměrně otevřený problém, a to i v pouhé definici pojmu anomálie. Problémem současných detektorů je velmi často jejich přílišná závislost na konkrétní scéně a nemožnost nasazení bez trénovacích dat [20]. Bylo identifikováno několik principiálně odlišných přístupů, pomocí kterých lze danou úlohu řešit. Důkladně byly prozkoumány i vhodné modely pozadí, jejichž možnost on-line aktualizace je pro řešenou úlohu zcela zásadní.

Následně byla v kapitole 3 provedena důkladná specifikace řešené úlohy z pohledu nasazení v reálných podmínkách a byly definovány přesné požadavky na řešení uživatelských úloh. Na základě teoretických znalostí byla navržena vhodná metoda stabilizace a dva principiálně odlišné detektory anomálie. Na základě teoretického návrhu vznikla specifikace konkrétní implementace z kapitoly 5. V kapitole 4 je proveden průzkum existujících datových sad a zhodnocení jejich relevantnosti z pohledu řešené úlohy. Na závěr jsou zde zdokumentovány veškeré experimenty s navrhovanými metodami, jejich vyhodnocení a porovnání s existujícími přístupy.

## Kapitola 2

# Detekce anomálie v davu

Detekce anomálie v davu je stále výrazně otevřeným problémem a to nejen v oblasti vývoje metod, ale i v pouhé interpretaci problému. Úlohu detekce anomálie lze rozdělit na globální a lokální [20]. U globální detekce je výstupem algoritmu pouhá informace o výskytu anomálie. Lokální detekce anomálie připojuje k globální detekci ještě informace o její poloze.

V odborných článcích [7, 28, 22] je za netypické chování davu obvykle považována změna rychlosti nebo směru pohybu, změnu vzhledu nebo výskyt netypického objektu. V posledních letech bylo v odborné literatuře publikováno hned několik přístupů k řešení této úlohy, jejichž shrnutí je včetně výběru konkrétních implementací uvedeno v kapitole 2.1.

S úlohou detekce anomálie úzce souvisí i metody výpočtu optického toku. Ten může být při řešení dané úlohy využit pro stabilizaci obrazu nebo pro popis pohybu ve scéně a je využíván velkou částí existujících přístupů [19, 41, 29]. Průzkum existujících metod výpočtu optického toku je uveden v kapitole 2.2. Problematika stabilizace obrazu je podrobněji rozebrána v kapitole 2.3.

Existující přístupy detekce anomálie velmi často reprezentují chování davu pomocí příznakových vektorů. Na základě extrahovaných příznaků je poté vytvářen statistický model obvyklého chování a detekce anomálie je zobecněna na detekci outlierů v příznacích [19]. Statistické modely se výrazně liší ve schopnostech relevantního popisu dat. V kontextu řešené úlohy je zásadní i možnost inicializace prázdného modelu a jeho následná iterativní aktualizace po dávkách. Přehled existujících přístupů ke statistickému modelování dat je uveden v kapitole 2.4.

Speciálním případem vytváření modelu chování davu nastává v případě využití rekurentních neuronových sítí, kde je zakódován v rámci stavu neuronové sítě a je tedy pevně spjat s její architekturou. Neuronové sítě mohou rovněž velmi často sloužit jako nástroj pro extrakci relevantních příznakových vektorů [41]. Přehled velmi často používaných architektur vzhledem k této problematice je uveden v kapitole 2.5.

### 2.1 Průzkum metod detekce anomálie

Část metod pro detekci abnormalit závisí na ručně získávaných příznacích pro reprezentaci pohybu nebo vzhledu davu. Články z posledních let ale poukazují na nesporné výhody využití neuronových sítí pro extrakci relevantního příznaku [30]. V případě malých datových sad jsou ale náchylné na přílišnou adaptaci na trénovací data a neschopnost dostatečného zobecnění. Nedostatek kvalitních datových sad lze považovat za jednu z největších překážek v řešení této úlohy [30, 20].

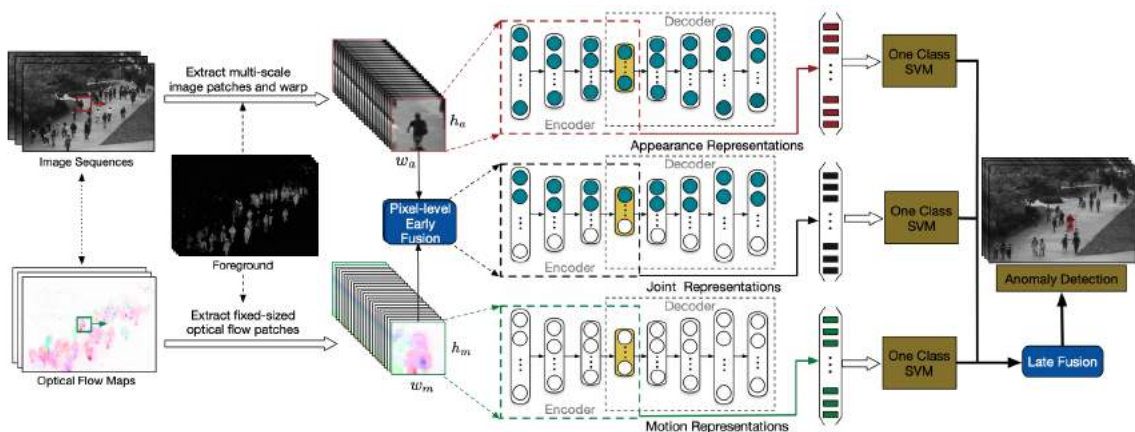
Přístupy detekce anomálie využívající učení bez učitele jsou oblastí, kterým je v posledních letech věnována velká pozornost zejména z důvodu nedostupnosti rozsáhlých datových sad. Jedná se o přístupy využívající zejména prediktivní nebo generativní modely [19]. Cílem prediktivních modelů je výpočet aktuálního snímku jako funkce několika snímků jemu předcházejících. Predikce aktuálního snímku je poté porovnána s reálnými daty a pomocí vhodné metriky je vyhodnocena úroveň anomálie. Generativní modely odhadují pravděpodobnostní rozdělení příznakových vektorů přicházejících dat a pro nový vzorek definují pravděpodobnost jeho výskytu.

Vzhledem k nedostatečné formalizaci celého problému lze na úlohu detekce anomálie rovněž nahlížet jako na detekci nežádoucího objektu ve scéně [7]. V tomto případě lze anomálii modelovat jako šablonu popisující nežádoucí objekt a využít k tomu obecné metody detekce.

## Analýza příznakových vektorů

Analýza příznakových vektorů je jedním z častých přístupů k úloze detekce anomálie. Hlavní myšlenkou je reprezentace optického toku pomocí relevantního příznakového vektoru a samotné řešení úlohy detekce anomálie spočívá v detekci outlierů porovnáním s dříve extrahovanými příznaky [25, 41, 19]. V některých článcích je referován ještě mezikrok v podobě shlukování příznakových vektorů a jejich následné kvantizace za účelem získání diskrétního identifikátoru [29]. Příznaky lze extrahovat pomocí klasických přístupů, ale v posledních letech věnuje odborná literatura větší pozornosti extrakci příznaků s využitím neuronové sítě z důvodu jejich větší robustnosti [20].

Jeden z nejnovějších přístupů detekce anomálie je publikován v článku *Learning Deep Representations of Appearance and Motion for Anomalous Event Detection* [41]. Tato metoda využívá kromě vstupní videosekvence ještě její variantu reprezentovanou v podobě optického toku a dále provádí i fúzi těchto dvou složek na úrovni jednotlivých pixelů. Videosekvence je rozdělena do mřížky tvořené buňkami. Pro snímek videosekvence, odpovídající optický tok a jejich fúzní kombinaci je pomocí neuronové sítě typu autoenkodér extrahován příznakový vektor. Na základě příznakových vektorů je poté detekována úroveň anomálie pomocí SVM klasifikátoru. Celková úroveň anomálie pro danou buňku je výsledkem kombinace výstupu analýzy pro jednotlivé složky. Ilustrace metody je na obrázku 2.1.



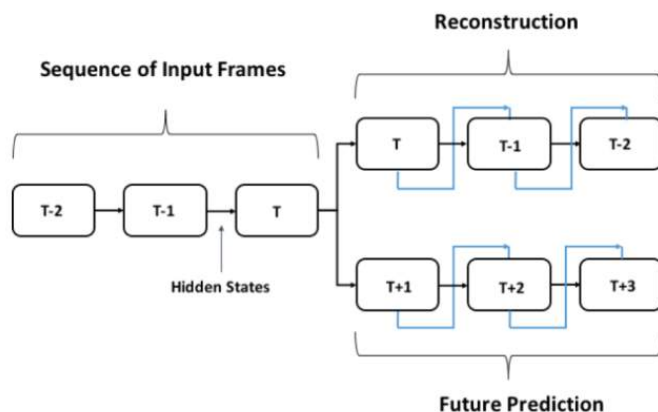
Obrázek 2.1: Ilustrace metody Learning Deep Representations of Appearance and Motion for Anomalous Event Detection (převzato z [41]).



## Metody využívající prediktivní modely

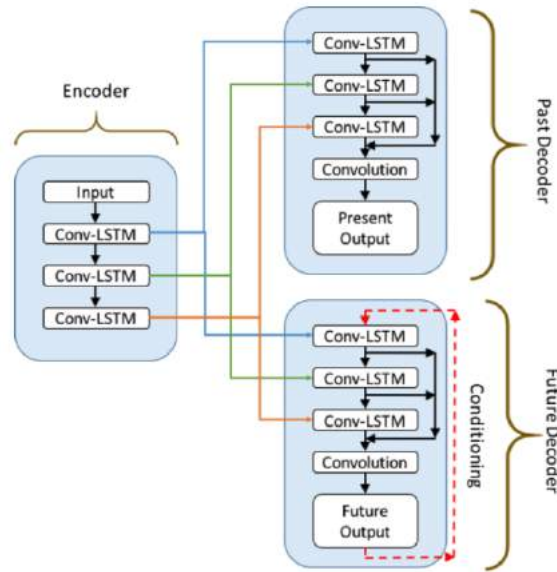
Cílem prediktivních modelů je modelování současného snímku jako funkce několika předcházejících snímků. Zástupcem prediktivních modelů jsou rekurentní neuronové sítě (RNN), které modelují tuto funkci jako rekurentní vztah zahrnující aplikaci nelinearity (např. sigmoidy). Velmi často používanou rekurentní neuronovou sítí pro zpracování sekvencí obrazových dat jsou neuronové sítě LSTM [19]. Úroveň anomálie bývá v tomto případě vypočtena na základě odchylky predikce a reálného snímku.

K hlubšímu prozkoumání jsem si vybral poměrně rozšířenou metodu využívající architekturu LSTM (*Long short-term memory*). Tato architektura umožňuje uchování časové závislosti v rámci poměrně dlouhé sekvence vstupních dat. Přístup z článku *Unsupervised learning of video representations using LSTMs* [35] nejprve extrahuje příznak ze vstupní sekvence několika snímků. Na základě tohoto příznaku poté odvozuje zpětnou rekonstrukci a predikci tohoto příznaku v čase. Přístup je znázorněn na obrázku 2.2. Při trénování je tedy cílem rekonstrukční sítě minimalizovat vzdálenost rekonstruovaného příznaku snímku v čase  $T - 2$  se skutečným příznakem. Cílem predikční sítě je minimalizace vzdálenosti mezi predikovaným příznakem v čase  $T + 3$  se skutečným příznakem. Úroveň anomálie poté odpovídá vzdálenosti predikovaných příznaků s reálně extrahovanými příznaky. Reálný příznakový vektor snímků je extrahován pomocí architektury autoenkodér. Proces trénování nevyžaduje žádné anotace trénovacích dat.



Obrázek 2.2: Ilustrace principu detekce anomálie pomocí metody založené na neuronové síti LSTM (převzato z [26]).

Na myšlenku předchozí metody staví metoda využívající architekturu *Composite Conv-LSTM Encoder-Decoder* [26]. Výhodou této metody je detekce anomálie pouze ve výřezích, které mají prostorovou souvislost. Zjednodušený náhled metody je na obrázku 2.3. Neuronová síť se skládá z několika menších sítí složených z konvolučních LSTM vrstev. Celkově lze neuronovou síť rozdělit na kodér a dekodér. Kodér očekává chronologickou posloupnost vstupní sekvence složenou z výřezů bez vzájemného překryvu. Výstupy každé z konvolučních LSTM vrstev jsou po průchodu posledního snímku vstupní sekvence využity jako příznaky. Každý z dekodérů je inicializován s příznaky odpovídajících vrstev kodéru. Výstup dekodéru je určen výhradně na základě počáteční inicializace. Výstup z každé vrstvy je konkatenován a sečten pomocí konvolučního jádra o velikosti  $1 \times 1$ . Tím je vypočtena rekonstrukce vstupu. Úroveň anomálie v daném snímku je opět určena na základě odchylky od predikce.

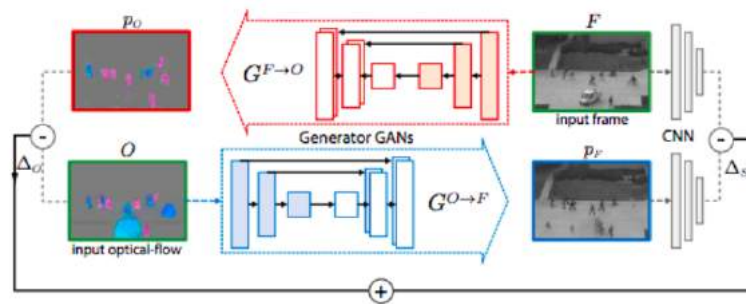


Obrázek 2.3: Blokový náhled architektury Composite Conv-LSTM Encoder-Decoder (převzato z [26]).

## Metody využívající generativní modely

Do kategorie metod využívajících generativní modely spadá zajímavá metoda publikovaná v článku *Abnormal Event Detection in Videos Using Generative Adversarial Nets* [30]. Je zde popisována metoda využívající neuronových sítí typu GAN. Tyto neuronové sítě jsou založeny na soutěžení dvou podsítí trénovaných na datech bez anotací. Jednou z nich je generátor usilující o generování realistických dat. Druhou sítí je diskriminátor, jehož úkolem je rozlišovat mezi uměle generovanými a reálnými daty.

Neuronová síť typu GAN je natrénována na snímcích bez výskytu anomálie v kombinaci s jejich variantou v podobě optického toku. Vzhledem k tomu, že se neuronová síť při procesu trénování nesetká s výskytem anomálie, není schopna ji ani vygenerovat. Během procesu detekce jsou porovnávána reálná a uměle vygenerovaná data a každá odlišnost je považována za anomálii. Schéma algoritmu této metody je na obrázku 2.4.



Obrázek 2.4: Schéma algoritmu metody využívající architektury GAN pro detekci anomálie (Převzato z [30]).

## Shrnutí

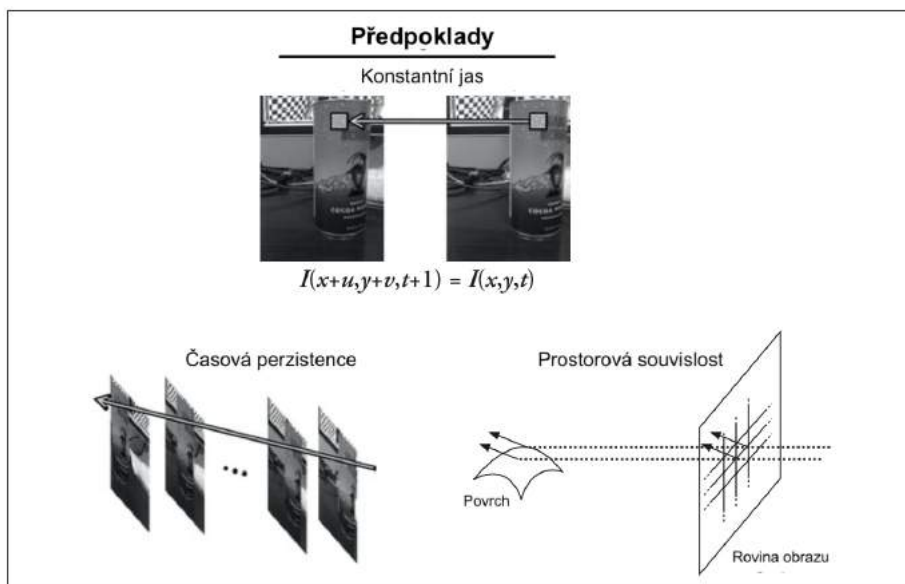
Během průzkumu existujících metod pro detekci anomálie v davových scénách se podařilo identifikovat několik rozdílných přístupů k řešení dané úlohy. Za vhodný přístup považuji extrakci relevantního příznaku optického toku a průběžné vytváření modelu jeho pravděpodobnostního rozložení. Tento přístup je nezávislý na konkrétní scéně a podobným řešením se již v minulosti podařilo dosáhnout uspokojivých výsledků [41]. Potenciál k řešení této úlohy mají i rekurentní neuronové sítě, nicméně spíše v oblasti extrakce příznakového vektoru. Uchovat informaci o obvyklém chování davu v rámci několikahodinového videozáznamu do vnitřního stavu sítě není příliš reálné [35]. Přístup využívající neuronové sítě typu *Generative Adversarial Nets* uvádí jejich závislost na konkrétní scéně, což je pro řešenou úlohu nevhodné [30].

## 2.2 Výpočet optického toku

Optický tok slouží k popisu změn obrazových dat v čase způsobených pohybem objektů ve scéně. Výsledkem metody výpočtu hustého optického toku je 2D pole vektorů, kde každý vektor reprezentuje směr a rychlost pohybu každého pixelu vstupních dat [34].

Metody výpočtu hustého optického toku jsou založeny na dvou základních předpokladech. Hodnota jasové funkce libovolného objektu ve scéně musí být konstantní vzhledem k času. Metody dále předpokládají, že se sousední body pohybují podobným způsobem [34].

Reálné scény nemusí vždy splňovat výše uvedené předpoklady, což je pro výpočet hustého optického toku komplikací. Z tohoto důvodu může být v některých případech využito metod realizujících výpočet řídkého optického toku [6]. Jedná se o alternativní přístup, jehož výsledkem je pohybový vektor definovaný pouze pro určitou podmnožinu vstupních dat. Metody spadající do této kategorie jsou založeny na detekci vhodných bodů ve scéně a jejich následném sledování v průběhu času. Jednou z nejpobulárnějších metod pro výpočet řídkého optického toku je metoda Lucas-Kanade [2].



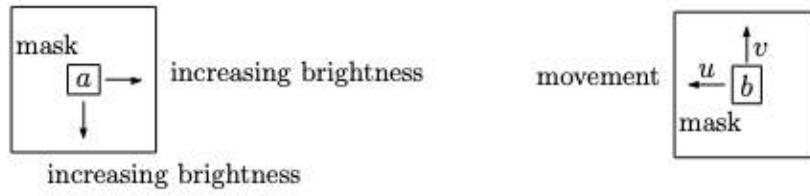
Obrázek 2.5: Předpoklady nutné ke správné funkci Lucas-Kanade algoritmu. (Převzato z [6])

## Lucas-Kanade metoda

Algoritmus pro výpočet řídkého optického toku Lucas-Kanade [2] byl poprvé publikován v roce 1981. Jedná se o jednoduchou techniku odhadující pohyb bodu zájmu v sousedních snímcích videa, kterou lze využít například pro úlohu stabilizace obrazu. Algoritmus požaduje kromě konstantní hodnoty jasové funkce a podobného pohybu sousedních bodů ještě omezení na maximální povolenou rychlost objektů ve scéně vzhledem k jejich měřítku. Předpoklady nutné pro správnou funkci algoritmu jsou ilustrovány na obrázku 2.5.

Předpokládejme sledování scény skrze rastr s pixelem **a**, což ilustruje obrázek 2.6. Intenzita pixelu **a** se v průběhu času mění a v následujícím snímku se změnila na intenzitu **b**. Lze tedy předpokládat, že došlo k posunutí sledovaného objektu směrem vlevo nahoru, díky čemuž je nová intenzita **b** viditelná. Známe-li gradient jasové funkce obrázku v osách  $x$  a  $y$ , platí pro pohyb objektu následující vztah 2.1, kde  $I_x$  je změna jasové funkce v ose  $x$ ,  $I_y$  je změna jasové funkce v ose  $y$  a  $I_t$  je změna času [32]. Cílem algoritmu je najít nejlepší odhad parametrů posunu  $\mathbf{u}$  a  $\mathbf{v}$ . Algoritmus k tomu v praxi využívá okolí o rozměrech  $3 \times 3$  pixely a parametry odhaduje na základě metody nejmenších čtverců.

$$I_x(x, y) \cdot u + I_y(x, y) \cdot v = -I_t(x, y) \quad (2.1)$$



Obrázek 2.6: Ilustrace k výkladu principu metody Lucas-Kanade. (Převzato z [32])

## Farnebackova metoda

Metoda [10] publikovaná v roce 2003 Gunnarem Farnebackem je spolehlivou a efektivní metodou výpočtu hustého optického toku. V prvním kroku provádí aproximaci okolí každého pixelu pomocí kvadratické funkce. Pixely prvního snímku  $f_1$  lze tedy vyjádřit vztahem 2.2, kde  $\mathbf{A}$  je symetrická matice,  $\mathbf{b}$  je vektor a  $c$  skalární hodnota. Jedná se o maticový zápis kvadratického polynomu. Posun těchto pixelů do následujícího snímku  $f_2$  je poté vyjádřen rovnicí 2.3, kde  $\mathbf{d}$  představuje vektor posunutí. Z tohoto systému rovnic lze poté vyjádřit hledaný vektor posunu  $\mathbf{d}$ . V praxi je do výpočtu zahrnuto i okolí zkoumaných bodů a další techniky zlepšující stabilitu metody. Ukázka hustého optického toku vypočteného pomocí Farnebackovy metody je na obrázku 2.7.

$$f_1(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c \quad (2.2)$$

$$f_2(\mathbf{x}) = f_1(\mathbf{x} + \mathbf{d}) = (\mathbf{x} + \mathbf{d})^T \mathbf{A} (\mathbf{x} + \mathbf{d}) + \mathbf{b}^T (\mathbf{x} + \mathbf{d}) + c \quad (2.3)$$

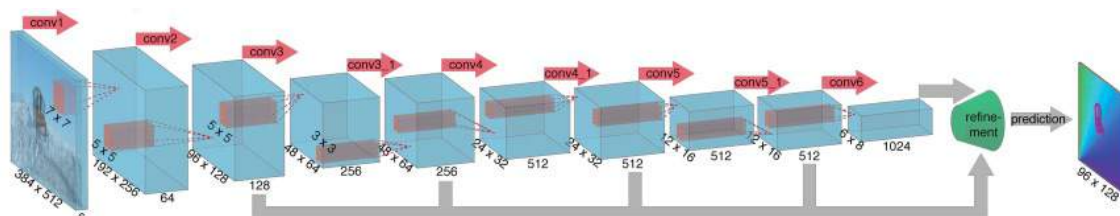


Obrázek 2.7: Ukázka hustého optického toku vypočteného pomocí Farnebackovy metody. (Převzato z [6])

## FlowNet

V posledních letech bylo dosaženo mnoha úspěchů v řešení širokého spektra úloh počítačového vidění pomocí neuronových sítí. Neuronová síť FlowNet [9] představuje jednu z prvních úspěšně navržených neuronových sítí pro výpočet optického toku. Základní architektura této neuronové sítě je na obrázku 2.8.

Vstupem neuronové sítě je dvojice po sobě jdoucích obrázků a jejím výstupem je vypočtený optický tok. Neuronová síť byla trénována za přítomnosti učitele. Vytvořit datovou sadu s anotací definující pohyb každého pixelu je velmi náročné. Pro trénování této neuronové sítě bylo z tohoto důvodu využito z velké části uměle renderovaných scén. Tato neuronová síť má mnoho nástupců, které dále vylepšují její přesnost a snižují výpočetní náročnost.



Obrázek 2.8: Architektura neuronové sítě FlowNet. (Převzato z [9])

## 2.3 Metody stabilizace obrazu

Úlohou stabilizace obrazu ve smyslu řešení úlohy je nalezení geometrické transformace mezi souřadnicovými systémy dvojicí snímků vstupní videosekvence. Základem pro nalezení geometrické transformace je identifikace dvojic odpovídajících bodů v rámci daných snímků. Na základě těchto dvojic lze pomocí vhodné metody nalézt aproximaci hledané transformace [6].

Tyto body musí splňovat určité vlastnosti a v odborné literatuře jsou označovány jako klíčové body. Jedním z možných přístupů pro vyhledání dvojic odpovídajících klíčových bodů je jejich sledování pomocí metody řídkého optického toku [6]. Nicméně, výrazným omezením tohoto přístupu je hranice pro maximální posun sledovaných bodů a nižší přesnost výsledku.

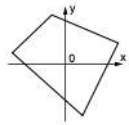
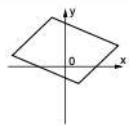
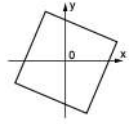
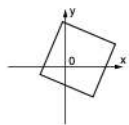
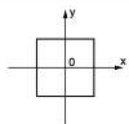
Druhým přístupem je extrakce příznakových vektorů popisujících klíčové body v obou snímcích a jejich následné propojení na základě porovnání příznaků [6]. Existuje množství metod určených pro extrakci příznakového vektoru klíčového bodu. Některé z těchto metod jsou robustní a dosahují velmi spolehlivých výsledků, nicméně jejich omezením je výpočetní náročnost.

## Geometrické transformace

Geometrický objekt je vždy reprezentován pomocí svých vrcholů. Geometrická transformace [34] je zobrazení definující obraz těchto bodů v modifikovaném souřadnicovém systému. Pro vyjádření souřadnic bodů se v počítačové grafice obvykle využívá tzv. homogenních souřadnic. Důvodem pro jejich zavedení je možnost snadného zápisu geometrické transformace pomocí transformační matice. Bod o souřadnicích  $[x, y]^T$  v kartézském souřadnicovém systému je pomocí homogenních souřadnic reprezentován vektorem  $[x, y, w]^T$  a pro jejich vzájemný převod platí rovnice 2.4, kde se používá  $w = 1$  pro reprezentaci bodů v 2D euklidovském prostoru a  $w = 0$  pro vektory.

$$[x, y]^T = [x/w, y/w]^T \quad (2.4)$$

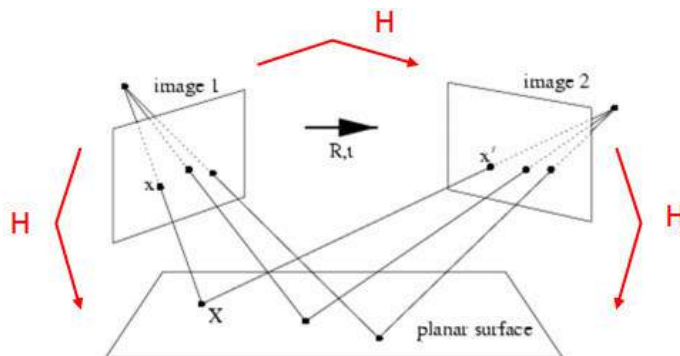
Geometrické transformace se rozdělují podle tvaru transformační matice do několika skupin [34], které jsou shrnuty v tabulce na obrázku 2.9. Základní skupinu geometrických transformací představují afinní transformace. Patří sem operace posunutí, rotace, změny měřítka nebo zkosení. Důležitou vlastností afinních transformací je zachování rovnoběžnosti.

Name	Constraints on $H$	2D example	Invariants
projective	$\det H \neq 0$		collinearity tangency cross ratio
affine	$H = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ $\det A \neq 0$		projective invariants + parallelism + length ratio on parallels + area ration + linear combinations of vectors centroid
similarity	$H = \begin{bmatrix} sR & -R\mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ $R^\top R = I$ $\det R = 1$ $s > 0$		affine invariants + angles + ratio of lengths
metric (Euclidean, isometric)	$H = \begin{bmatrix} R & -R\mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$ $R^\top R = I$ $\det R = 1$		similarity invariants + length + area (volume)
identity	$H = I$		trivial case everything is invariant

Obrázek 2.9: Shrnutí geometrických transformací. (Převzato z [34])



Nicméně pomocí afinních transformací není možné vyřešit problém mapování souřadnicových systémů dvojice obrázků pořízených pohyblivou kamerou, což je ilustrováno schématickým zobrazením 2.10. Geometrickou transformací popisující toto zobrazení je homografie [6]. Homografii ve 2D prostoru lze popsat pomocí transformační matice s rozměry  $3 \times 3$  a celkem 8-mi neznámými ( $H_{33} = 1$ ). Pro výpočet matice popisující homografii jsou tedy potřeba nejméně 4 dvojice odpovídajících bodů.



Obrázek 2.10: Schématické zobrazení geometrického modelu pro hledání homografie mezi dvojicí obrázků stejné scény. (Převzato z [6])

Při řešení úloh stabilizace obrazu je obvykle dostupno mnohem více dvojic odpovídajících bodů, než jejich minimální počet. Požadavkem na metodu hledání homografie je tedy její robustnost vzhledem k případným outlierům. Velmi často používanou metodou pro hledání homografie splňující požadavky robustnosti je metoda RANSAC (*Random Sample Consensus*) [6].

Metoda RANSAC [12] provádí náhodný výběr odpovídajících  $N$ -tic bodů. Na jejich základě provede výpočet matice homografie. V další kroku určí podíl inlierů z celkové množiny dvojic bodů vzhledem k nalezené homografii. Tyto kroky jsou prováděny v omezeném počtu iterací. Poté je vybrána varianta s největším podílem inlierů a podle nich je pomocí metody nejmenších čtverců vypočtena výsledná aproximace matice homografie.

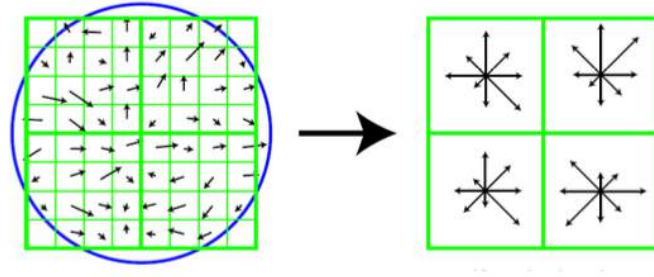
## Klíčové body a jejich deskriptory

Klíčové body v obrázku definují souřadnice vizuálně významných míst vhodných k jejich analýze. Ke klíčovým bodům mohou být připojeny deskriptory. Jedná se o příznakový vektor jednoznačně popisující daný klíčový bod a jeho blízké okolí. U deskriptoru klíčového bodu je zásadní jeho robustnost vůči geometrickým transformacím nebo vizuálním transformacím. Metody detekce klíčových bodů a extrakce deskriptorů se liší v parametrech robustnosti i výpočetní náročnosti. Z důvodu volby vhodné metody pro řešení uživatelské úlohy byl proveden přehled těch nejznámějších přístupů [34].

Metoda SIFT (*Scale Invariant Feature Transform*) [21] slouží k extrakci stabilních klíčových bodů společně s jejich robustními deskriptory. Metoda využívá vlastnosti Laplaceanu Gaussovských rozložení (LoG), který se chová jako detektor skvrn o různých velikostech, čehož je docíleno změnou parametru  $\sigma$ . Výpočet Laplaceanu je ale výpočetně náročná operace a z tohoto důvodu jej algoritmus aproximuje pomocí rozdílu Gaussovských rozložení (DoG).

Vztah pro výpočet rozdílu Gaussovských rozložení je vyjádřen rovnicí 2.5, kde  $I$  je jasová funkce zkoumaného obrázku a  $G$  je Gaussovské jádro s variancí  $\sigma$ . Výpočet probíhá napříč různými rozlišeními obrázku, čímž je zajištěna invariance vzhledem ke změně měřítka. Skvrny jsou v rozdílu Gaussovských rozložení detekovány jako lokální extrémy.

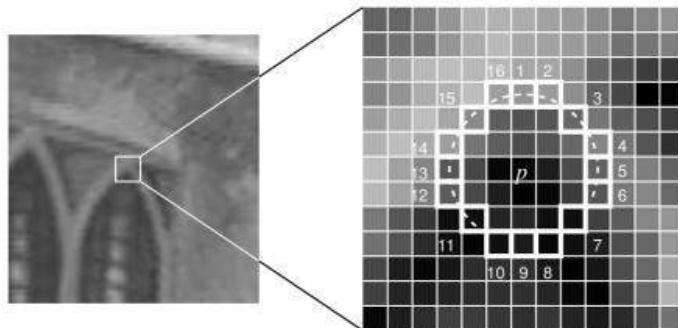
$$D(\mathbf{x}, \sigma) = (G(\mathbf{x}, k\sigma) - G(\mathbf{x}, \sigma)) * I(\mathbf{x}) \quad (2.5)$$



Obrázek 2.11: Obrázek ilustrující výpočet deskriptoru pomocí metody SIFT. (Převzato z [21])

Deskriptor detekovaného klíčového bodu vypočtený metodou SIFT [21] představuje histogram gradientů jasové funkce pro jednotlivé pixely a je ilustrován obrázkem 2.11. Pro výpočet deskriptoru klíčového bodu je využito okolí o celkových rozměrech 8x8 pixelů. Toto okolí je dále rozděleno do menších buněk o rozměrech 4x4 pixely a pro každou z nich je sestaven histogram rozložení vektorů gradientu. Příspěvek vektorů gradientu do histogramu klesá se vzdáleností od souřadnic klíčového bodu. Vysoká úspěšnost této metody je vykoupena vysokou výpočetní náročností. Příznaky SIFT není možné extrahovat na videu v reálném čase.

Velmi dobrou alternativu k metodě SIFT z pohledu redukované výpočetní náročnosti představuje metoda ORB (*Oriented FAST and Rotated BRIEF*) [33]. Metoda ORB spojuje detektor klíčových bodů FAST a extraktor příznakového vektoru BRIEF. Funkce detektoru FAST je ilustrována obrázkem 2.12. Detektor nejprve vyhledává body v obrázku splňující určité podmínky definující body zájmu. Intenzita tohoto bodu je porovnávána s 16-ti body v jeho okolí. Bod je považován za rohový, pokud se v kruhu nachází n po sobě jdoucích pixelů, jejichž jasová funkce se liší od jeho hodnoty jasové funkce o stanovený práh.

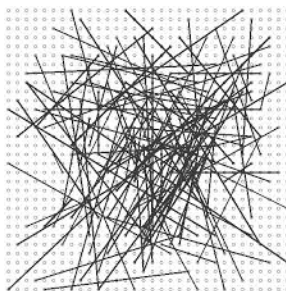


Obrázek 2.12: Obrázek ilustrující funkci detektoru klíčových bodů FAST. (Převzato z [6])



Deskriptor metody SIFT představuje příznak o celkovém počtu 128 dimenzí. Nicméně všechny tyto dimenze nemusí být z pohledu porovnávání deskriptorů relevantním měřítkem [6]. LSH (*Local sensitive hashing*) metoda a jí podobné umožňují převod reálných čísel tohoto deskriptoru do binárního řetězce. Vzdálenost binárních řetězců lze snadno a výpočetně efektivně vyjádřit pomocí Hammingovy vzdálenosti.

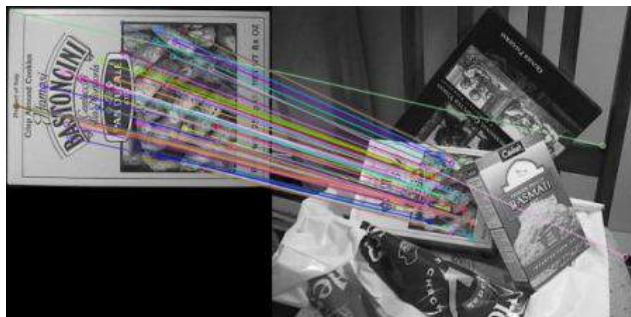
Metoda BRIEF [33] extrahuje výše zmíněné binární řetězce bez nutnosti vyjádření deskriptoru pomocí reálných čísel. Výpočet binárního příznaku je založen na porovnávání dvojic pixelů ilustrovaných obrázkem 2.13. Každý bit extrahovaného příznaku reprezentuje pravdivostní hodnotu výsledku porovnání odpovídající dvojice pixelů. Výsledný ORB detektor do těchto přístupů přináší ještě další optimalizace z pohledu robustnosti a výpočetní náročnosti.



Obrázek 2.13: Obrázek ilustrující dvojice bodů, na jejichž základě je vytvořen binární příznak metody BRIEF. (Převzato z [6])

Hledání klíčových bodů a jejich deskriptorů může mimo jiné posloužit i jako základ pro hledání homografie mezi dvojicí transformovaných obrázků téže scény. Při hledání matice homografie je zásadní nalezení bodů, které si v jednotlivých obrázcích odpovídají. Základním algoritmem pro řešení této úlohy je Brute-Force matcher [6]. Každý deskriptor náležící množině bodů prvního obrázku je porovnán s každým deskriptorem bodů množiny druhého obrázku. Algoritmus vrací dvojice bodů s nejmenší vzdáleností určené metrikou vhodnou pro daný deskriptor. Je nevýhodou je vysoká výpočetní náročnost.

Výpočetně optimalizovaným ekvivalentem Brute-Force matcheru je metoda FLANN (*Fast Library for Approximate Nearest Neighbors*) based matcher [27]. Tato metoda obsahuje kolekce algoritmů optimalizovaných pro rychlé hledání nejbližšího souseda ve velkých datasetech. K akceleraci vyhledávání je využita datová struktura k-D strom. Ilustrace očekávaného výstupu metody pro hledání dvojic korespondujících bodů je na obrázku 2.14.



Obrázek 2.14: Výstup Brute Force matcheru při hledání korespondujících dvojic bodů na základě SIFT příznaků. (Převzato z [6])

V některých případech není výpočet deskriptoru klíčového bodu při hledání matice homografie potřebný. Jedním z těchto případů je hledání korespondujících klíčových bodů dvojice snímků na základě optického toku. Výpočetně efektivním přístupem pro detekci samostatných klíčových bodů v obraze je Harrisův detektor.

Harrisův detektor [8] je matematický operátor určený pro hledání rohů v obraze. Roh se v obraze definovaném jasovou funkcí  $I(x, y)$  nachází v bodě  $(x, y)$  právě tehdy, když v okolí tohoto bodu dochází k výrazné změně jasové funkce ve všech směrech posunu  $(\Delta x, \Delta y)$ . Porovnání změny jasové funkce neprobíhá pouze na základě jediného pixelu, ale v rámci malého okna definovaného funkcí  $w(x, y)$ . Matematicky lze tuto formulaci vyjádřit vztahem 2.6.

Posunování pohyblivého okna v rámci obrazu je výpočetně náročnou operací. Z tohoto důvodu byla na základě původní rovnice 2.6 odvozena její aproximace aplikací Taylorova rozvoje a několika dalších úprav. Aproximace je vyjádřena vztahem 2.7, kde  $I_x$  je derivace jasové funkce podle osy  $x$  a  $I_y$  je derivace jasové funkce podle osy  $y$ .

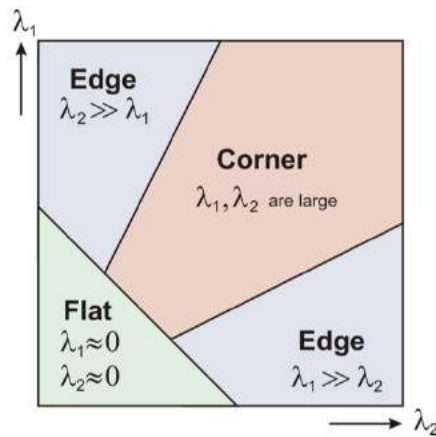
$$E(\Delta x, \Delta y) = \sum_{x,y} \underbrace{w(x, y)}_{\text{window function}} \underbrace{[I(x + \Delta x, y + \Delta y) - I(x, y)]^2}_{\substack{\text{shifted intensity} \\ \text{intensity}}} \quad (2.6)$$

$$E(\Delta x, \Delta y) \approx \begin{bmatrix} \Delta x & \Delta y \end{bmatrix} \left( \sum_{x,y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (2.7)$$

$$M = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (2.8)$$

$$R = \lambda_1 \lambda_2 - k \cdot (\lambda_1 + \lambda_2)^2 \quad (2.9)$$

Vlastní čísla  $\lambda_1$  a  $\lambda_2$  matice  $M$  ze vztahu 2.8 reflektují vliv jednotlivých směrů na variabilitu jasové funkce v okolí zkoumaného bodu a podle jejich hodnot lze určit charakter struktury detekované v daném místě. Vysoké hodnoty obou vlastních čísel znamenají vysokou variabilitu jasové funkce ve všech směrech, což indikuje výskyt rohu. Další případy jsou ilustrovány obrázkem 2.15. Pro daný bod je vypočteno skóre podle rovnice 2.9 a v případě překročení určitého prahu jej považujeme za roh.



Obrázek 2.15: Význam hodnot vlastních čísel počítaných při detekci rohů pomocí Harrisova detektoru. (Převzato z [6])

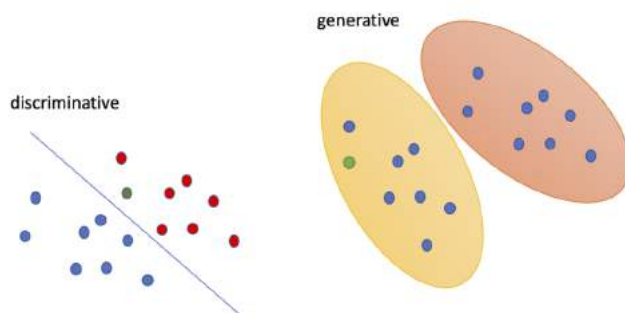
## 2.4 Strojové učení

Cílem strojového učení je proměna trénovacích dat v užitečnou informaci [6]. Toho je dosaženo extrakcí pravidel nebo vzorů, které se v datech nacházejí. Metody strojového učení mohou pracovat nad libovolnými daty reprezentovatelnými v digitální doméně. Pro příklad lze uvést teplotu, ceny komodit na burzách nebo fotografie. Data jsou obvykle transformována do příznakových vektorů. Příznakový vektor je výsledkem nějakého pozorování a lze si jej představit jako bod v souřadnicovém prostoru příznaků. Algoritmus strojového učení vytváří na základě příznakových vektorů jejich statistický model, na základě kterého dokáže řešit klasifikační nebo regresní úlohy.

Algoritmy strojového učení lze rozdělit do dvou kategorií - s učitelem a bez učitele [6]. Trénovací data pro algoritmy strojového učení s učitelem obsahují pro každý trénovací vzorek rovněž štítek definující očekávaný výstup metody. Štítek může definovat kategorii příslušnosti daného vzorku. V tomto případě se jedná o klasifikační úlohu. Pokud je očekávaným výstupem algoritmu reálná hodnota, jedná se o regresní úlohu. V reálných úlohách ale velmi často nastává situace, kdy nejsou žádné štítky trénovacích dat dostupné a odhalení vzorů v trénovacích datech je čistě na algoritmu učení bez přítomnosti učitele. Cílem těchto metod je redukce dimenzí, statistická analýza dat nebo identifikace podobných vzorků a jejich rozdělení do jednotlivých tříd.

Velmi dobrých výsledků v úlohách klasifikace, regrese i clusterizace je v současnosti dosahováno pomocí přístupů založených na neuronových sítích. Jejich podrobnější rozbor je uveden v kapitole 2.5. Nevýhodou neuronových sítí oproti klasickým přístupům je potřeba rozsáhlého trénovacího datasetu. Toto představuje limitaci jejich využití vzhledem ke kontextu řešené úlohy [15]. Pomocí vhodného statistického přístupu lze nalézt uspokojivé řešení i pro úlohy, které by z důvodu nedostatku trénovacích dat nebylo možné pomocí neuronových sítí vyřešit.

Statistické modely trénovacích dat lze principiálně rozdělit na diskriminativní a generativní [6]. Diskriminativní modely se zaměřují na vymezení rozhodovací hranice mezi třídami klasifikace. Z pohledu statistiky modelují pravděpodobnost kategorie na základě vzorku dat, což lze matematicky zapsat vztahem  $p(class|sample)$ . Oproti tomu, generativní modely se zaměřují na popis dat v rámci jediné třídy a modelují pravděpodobnost  $p(sample|class)$ . Rozdíl těchto dvou přístupů je vyjádřen na obrázku 2.16.



Obrázek 2.16: Obrázek ilustrující rozdíly mezi generativními a diskriminativními statistickými modely. (Převzato z webu<sup>2</sup>)

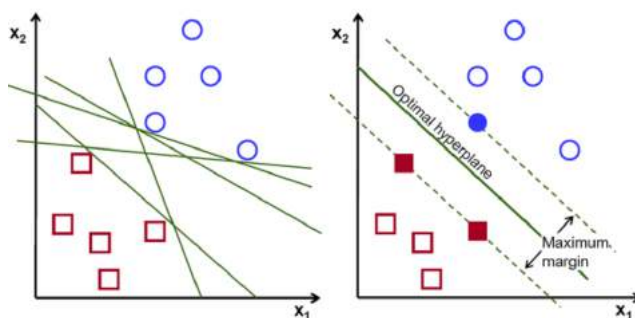
<sup>2</sup><http://mbenhaddou.com/2019/12/13/generative-and-discriminative-models/>

## Support vector machine (SVM)

*Support vector machine* [34] je algoritmus strojového učení určený pro úlohu binární klasifikace. Jeho cílem je nalezení nadroviny oddělující množiny vzorků dvou různých tříd. Nadrovina dimenze  $N$  je definována jako její libovolný podprostor s dimenzí  $N - 1$ . Příznakové vektory se označují pojmem support vectors a definují předpis diskriminační funkce popisující hraniční nadrovinu.

V praxi mohou nastat situace, kdy je nutné nalézt diskriminační funkci rozdělující množiny lineárně neseparovatelných dat. V tomto případě lze transformovat body původního souřadnicového systému do prostoru, ve kterém jsou již lineárně separovatelné. Tento proces je označován pojmem jádrová transformace.

Výhodou tohoto algoritmu oproti lineárnímu klasifikátoru je garance robustnější rozhodovací hranice. Rozdíl mezi rozhodovací hranicí nalezenou pomocí lineárního a SVM klasifikátoru je ilustrován na obrázku 2.17. Řešení nalezené SVM klasifikátorem maximalizuje na rozdíl od lineárního klasifikátoru vzdálenost rozhodovací hranice od vzorků obou tříd.



Obrázek 2.17: Porovnání rozhodovacích hranic binárního klasifikačního problému nalezené pomocí lineárního a SVM klasifikátoru. (Převzato z webu<sup>4</sup>)

## K-Means

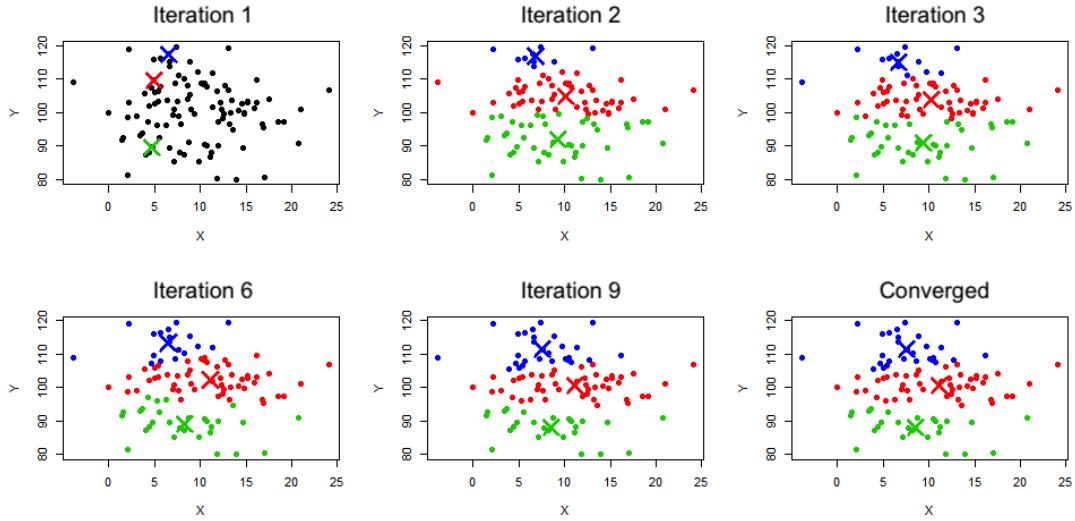
Cílem clusterizace je rozdělení množiny trénovacích vzorků do podmnožin na základě jejich podobnosti. Hierarchické metody clusterizace využívají stromových struktur pro rekursivní dělení výchozí množiny do různého počtu menších podmnožin. Tyto metody se obvykle využívají pro optimalizované trénování na velkých datasetech, on-line datech nebo v případech, kdy výsledný počet clusterů není předem známý [34]. Zástupcem hierarchických metod clusterizace je například metoda Birch [42].

Algoritmus K-Means [6] je populárním algoritmem strojového učení bez učitele pro řešení úlohy clusterizace. Metoda nevyužívá hierarchických datových struktur a počet clusterů je stanoven před spuštěním algoritmu. Pozice středu každého z clusterů je před spuštěním výpočtu náhodně inicializována.

Algoritmus iterativně opakuje dvojici kroků dokud nedojde ke konvergenci. V prvním kroku jsou trénovací vzorky rozděleny mezi jednotlivé clustery podle jejich vzdálenosti od středu. V druhém kroku jsou aktualizovány souřadnice středu clusterů na základě aktuálního rozdělení testovacích vzorků. Několik iterací algoritmu je ilustrováno obrázkem 2.18.

<sup>4</sup><https://towardsdatascience.com/svm-feature-selection-and-kernels-840781cc1a6c>

<sup>6</sup><http://www.learnbymarketing.com/methods/k-means-clustering/>



Obrázek 2.18: Ilustrace několika iterací algoritmu K-Means na 2D datech. (Převzato z webu<sup>6)</sup>)

## Směs Gaussovských rozložení

Gaussovo rozložení [31] je typ pravděpodobnostního rozložení hustoty pravděpodobnosti popisující spojité náhodné proměnné. Funkce hustoty rozdělení pravděpodobnosti tohoto rozložení je definována vztahem 2.10. Koeficient umístěný před exponenciální funkcí je zde pouze pro účely normalizace. Gaussovo rozdělení pravděpodobnosti je velmi často používáné z důvodu jeho přirozeného výskytu v náhodných jevech.

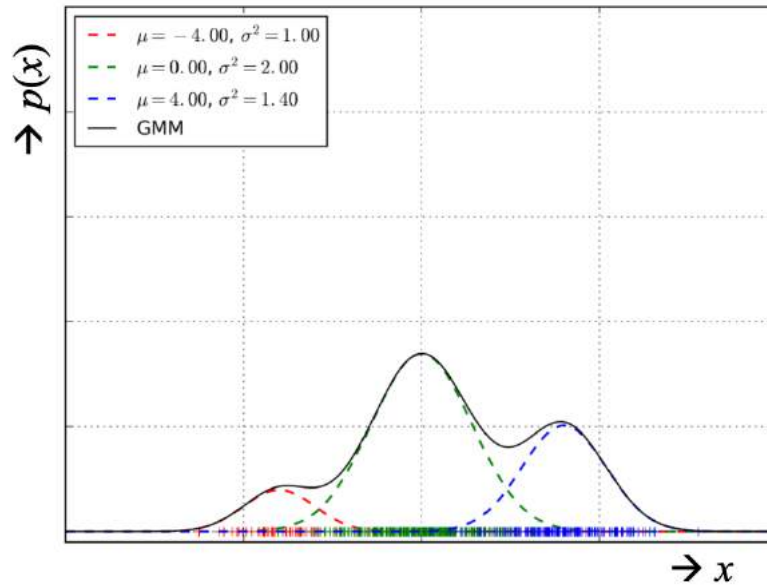
$$p(x) = \mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2.10)$$

Postačujícími statistikami pro popis sady vzorků pomocí Gaussovského rozložení je součet jejich druhých mocnin, lineární součet a jejich celkový počet. Pomocí Gaussovského rozložení je možné popisovat data o libovolném počtu dimenzí. Jejich statistická závislost je poté vyjádřena pomocí parametru kovarianční matice  $\Sigma$ , který nahrazuje parametr variance  $\sigma$ . Pro odhad parametrů Gaussovského rozložení vícedimenzionálních proměnných platí vzorec 2.11.

$$\begin{aligned} \boldsymbol{\mu} &= \frac{1}{N} \sum_n \mathbf{x}_n \\ \boldsymbol{\Sigma} &= \frac{1}{N} \sum_n (\mathbf{x}_n - \boldsymbol{\mu})(\mathbf{x}_n - \boldsymbol{\mu})^T \end{aligned} \quad (2.11)$$

Směs Gaussovských rozložení (GMM) [31] je pravděpodobnostní model popisující hustotu rozložení pravděpodobnosti spojitých náhodných proměnných pomocí váhovaného součtu klasických Gaussovských rozložení, což je ilustrováno obrázkem 2.19. Funkci hustoty rozdělení pravděpodobnosti pro GMM popisuje rovnice 2.12, kde  $x$  je vzorek dat a  $\eta$  symbolizuje parametry modelu. Každá komponenta směsi Gaussovských rozložení je specifikována pomocí střední hodnoty  $\mu_c$ , rozptylu  $\sigma_c$  a váhy  $\pi_c$ . Součet vah jednotlivých komponent je roven 1.

$$p(x | \boldsymbol{\eta}) = \sum_c \mathcal{N}(x; \mu_c, \sigma_c^2) \pi_c \quad (2.12)$$



Obrázek 2.19: Ukázka funkce rozdělení hustoty pravděpodobnosti pro Gaussian Mixture Model. (Převzato z webu<sup>8</sup>)

Pro trénování směsi Gaussovských rozložení se využívá obecně aplikovatelného přístupu Expectation Maximization [31]. Tento přístup garantuje zlepšení modelu v každé iteraci, nicméně nemusí vždy najít globální optimum. Parametry GMM jsou před spuštěním algoritmu inicializovány náhodně. Algoritmus provádí iterativní opakování dvojice kroků. V prvním kroku jsou vypočteny faktory odpovědnosti jednotlivých komponent za vygenerování vzorků dat podle vztahu 2.13. Na jejich základě jsou následně aktualizovány parametry GMM modelu podle vztahů 2.14. Algoritmus je ukončen v případě dosažení konvergence.

$$\gamma_{zi} = \frac{p(x_i | z_i) P(z_i)}{\sum_k p(x_i | z_i = k) P(z_i = k)} \quad (2.13)$$

$$\begin{aligned} \mu_k^{(new)} &= \frac{1}{\sum_i \gamma_{ki}} \sum_i \gamma_{ki} x_i \\ \sigma_k^{2(new)} &= \frac{1}{\sum_i \gamma_{ki}} \sum_i \gamma_{ki} (x_i - \mu_k^{(new)})^2 \\ \pi_k^{(new)} &= \frac{\sum_i \gamma_{ki}}{\sum_k \sum_i \gamma_{ki}} = \frac{\sum_i \gamma_{ki}}{N} \end{aligned} \quad (2.14)$$

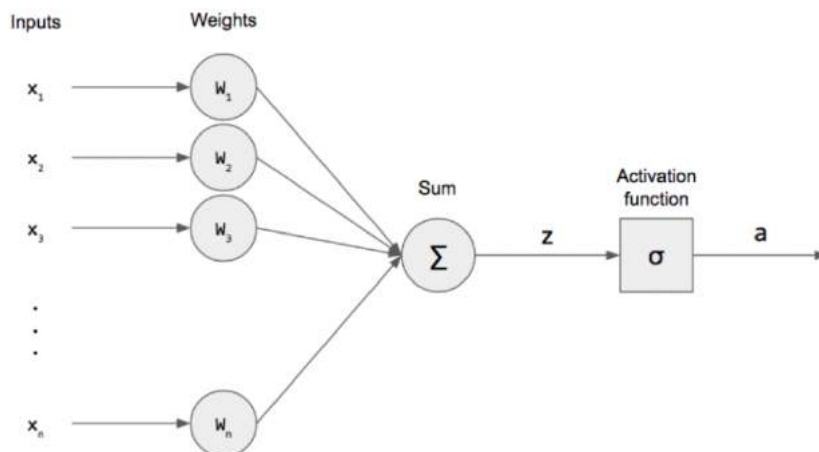
## 2.5 Architektury neuronových sítí

Neuronová síť [34] je matematický výpočetní model a kolekce algoritmů strojového učení. Jedná se o velmi dobrý nástroj pro řešení mnoha úloh nejen v oblasti počítačového vidění. Většina přístupů založených na tomto výpočetním modelu využívá mnoha elementárních jednotek nazývaných perceptrony. Perceptron představuje zjednodušený matematický model neuronu lidského mozku a je ilustrován na obrázku 2.20.

Perceptron [34] se skládá z několika vstupů vyjádřených vstupním vektorem  $\vec{x} = (x_0, x_1, \dots, x_n)$ . Každý vstup má k sobě přiřazenou váhu, které jsou souhrnně reprezentovány vektorem vah  $\vec{w} = (w_0, w_1, \dots, w_n)$ . Vnitřní potenciál perceptronu lze v případě lineární báze

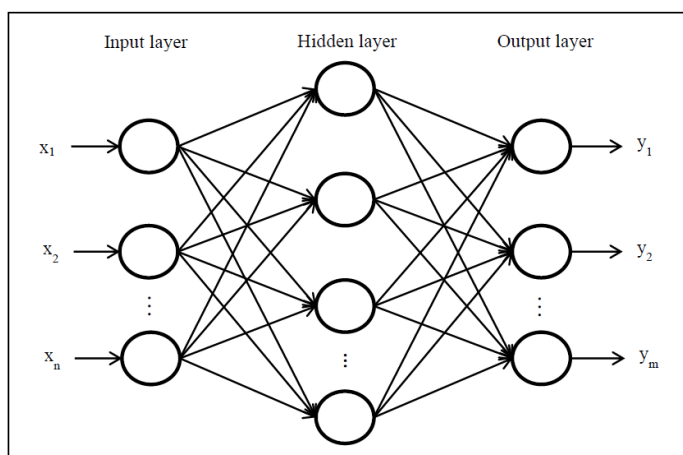
<sup>8</sup><https://www.fit.vutbr.cz/study/courses/BAYa/public/prednasky/.cs>

funkce vyjádřit jako  $z = \vec{x} \cdot \vec{w}$ . Výstup neuronu je poté určen aplikací aktivační funkce na hodnotu vnitřního potenciálu, matematicky tedy  $a = f(z)$ , kde  $f$  je aktivační funkcí. Perceptron umožňuje řešit pouze lineárně separovatelné úlohy, což vedlo ke konstrukci neuronových sítí.



Obrázek 2.20: Schéma ilustrující výpočetní model perceptronu. (Převzato z webu<sup>9</sup>)

Mezi základní architektury neuronových sítí patří jednoduchá dopředná neuronová síť [36] ilustrovaná obrázkem 2.21. Neurony jsou v rámci této sítě seskupovány po vrstvách. Výstup každého neuronu z předchozí vrstvy je propojen s každým neuronem následující vrstvy. Výpočet odezvy neuronové sítě probíhá sekvenčně po jednotlivých vrstvách. Pomocí maticového zápisu lze tedy vyjádřit vektor vnitřních potenciálů dané vrstvy pomocí vztahu  $\vec{a} = \vec{x} \cdot \mathbf{w}$ , kde  $\vec{x}$  je vektor vstupních hodnot a  $\mathbf{w}$  je matice vah. Vektor výstupních hodnot lze nyní vyjádřit vztahem  $\vec{y} = f(\vec{a})$ , kde  $f$  je aktivační funkce pracující s vektory.



Obrázek 2.21: Schéma uspořádání neuronů v jednoduché dopředné neuronové síti. (Převzato z webu<sup>11</sup>)

<sup>9</sup><https://pythonmachinelearning.pro/perceptrons-the-first-neural-networks/>

<sup>11</sup><https://www.researchgate.net/figure/The-structure-of-the-two-layered-feed-forward-neural-network>

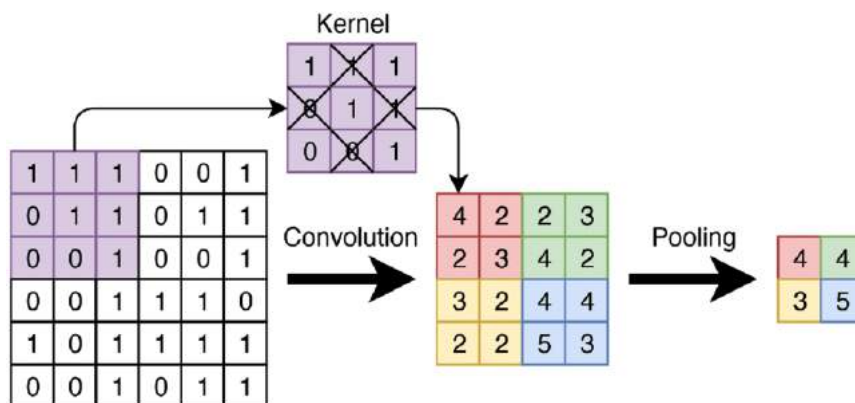


Cílem algoritmů pro trénování neuronové sítě je hledání optimálních vah, které minimalizují hodnotu chybové funkce pro trénovací datovou sadu. Trénovací datová sada je ve tvaru  $T = \{(\vec{x}_1, \vec{y}_1), (\vec{x}_2, \vec{y}_2), \dots, (\vec{x}_n, \vec{y}_n)\}$ , kde dvojice  $(\vec{x}_i, \vec{y}_i)$  představují vstup a očekávanou odezvu. Speciální dopřednou neuronovou sítí je architektura autoenkodér. Při jejím trénování se používají identické vektory vstupu a očekávané odezvy. Smyslem autoenkodéru je kódování důležitých rysů vstupních vektorů do nižších dimenzí a jejich zpětná rekonstrukce, čehož se často využívá při extrakci příznakových vektorů [3].

Hodnota chybové funkce vyjadřuje míru správnosti odezvy a definuje tedy korektní chování neuronové sítě. Jedním ze základních algoritmů pro nastavování vah je metoda gradientního sestupu. Myšlenkou tohoto přístupu je derivace chybové funkce podle vah neuronů a její minimalizace na základě sestupu v opačném směru proti vektoru gradientu. [36]

V neuronových sítích pro zpracování obrazových dat se velmi často vyskytují konvoluční nebo seskupující vrstvy. Důvodem k tomu je vizuální souvislost sousedních pixelů, kterou lze pomocí těchto vrstev zachytit [13]. Konvoluční vrstva realizuje operaci konvoluce vstupních dat s libovolným počtem konvolučních jader. Obvykle se používají konvoluční jádra o rozměrech 3x3, 5x5 nebo 7x7 pixelů a jejich hodnoty jsou určeny během procesu trénování.

Cílem seskupujících vrstev je snížení rozlišení při zachování relevantních informací extrahovaných konvoluční vrstvou. Vstupní data jsou v této vrstvě procházena pomocí pohyblivého okna s předem definovaným rozlišením a krokem posunu. Výsledná hodnota pro danou pozici okna je vypočtena operací maxima, aritmetického průměru nebo jinou seskupující funkcí. Funkce konvolučních a seskupujících vrstev je souhrnně ilustrována obrázkem 2.22.



Obrázek 2.22: Ilustrace funkce konvolučních a seskupujících vrstev v neuronových sítích. (Převzato z webu<sup>13</sup>)

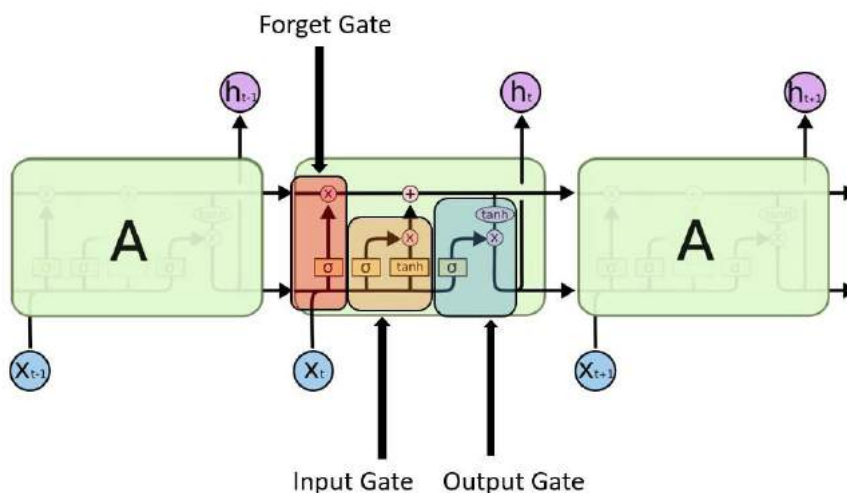
Specifikem obrazových dat v rámci videosekvence je kombinace prostorové i časové souvislosti. K zachycení časové souvislosti v rámci sekvence dat jsou velmi často používány rekurentní neuronové sítě [15]. Jedná se o třídu neuronových sítí, u které je při výpočtu odezvy využito kromě aktuálního i předchozích vstupů. Informace z předchozích vstupů je zakódována do vnitřního stavu sítě.

<sup>13</sup><https://www.researchgate.net/figure/Schematic-representation-of-a-convolution-and-pooling-layer-in-a-CNN>



Typickou ukázkou rekurentní neuronové sítě je bezesporu síť *LSTM* (*Long Short Term Memory*) [17]. Architektura této sítě je ilustrována na obrázku 2.23. Konkrétně je zde ilustrován výpočet odezvy  $h_{t-1}$ ,  $h_t$  a  $h_{t+1}$  pro tři po sobě následující vzorky vstupních dat  $x_{t-1}$ ,  $x_t$  a  $x_{t+1}$ . V každém časovém kroku jsou prováděny modifikace vnitřního stavu buňky pomocí jednotek *forget*, *input* a *output gate*.

Jednotka *forget gate* vyhodnocuje, která informace vnitřního stavu bude v daném kroku zapomenuta. Pomocí jednotky *input gate* je vybrána užitečná informace vstupního vektoru a následně je uložena do vnitřního stavu. Po modifikaci vnitřního stavu je vypočtena výsledná odezva buňky pomocí jednotky *output gate*. Vnitřní stav zůstává uchovaný pro další časový krok.



Obrázek 2.23: Schéma architektury rekurentní neuronové sítě LSTM. (Převzato z webu<sup>14</sup>)

<sup>14</sup><https://wiki.tum.de/display/lfdv/Recurrent+Neural+Networks+-+Combination+of+RNN+and+CNN>

## Kapitola 3

# Rozbor úlohy a její dekompozice

V kapitole 2 byl proveden důkladný průzkum existujících přístupů k analýze davových scén. Nyní bude nejprve proveden rozbor uživatelských úloh a případů užití, což jsou zcela zásadní informace pro návrh adekvátního uživatelského nástroje. Na základě načerpaných informací budou v úloze identifikovány dílčí podproblémy a bude navrženo jejich řešení společně s celkovou architekturou systému.

### 3.1 Rozbor uživatelské úlohy

Cílem uživatele je sledování mnohdy i rozsáhlého prostoru pomocí dronu s kamerou. Dron je po svém vzletu navigován do pozice vhodné pro sledování dané scény, která je ale omezena řadou bezpečnostních předpisů. Dron zaujímá stacionární polohu ve výšce 80 - 150 metrů nad terénem. Kamera poskytuje RGB videozáznam scény o typických rozměrech 300x300 metrů s rozlišením 1920x1080 pixelů. Ukázka snímků z kamery je na obrázku 3.1. Scéna může být snímána za nízkého osvětlení během ranních nebo večerních hodin. Dále je zde riziko narušení viditelnosti vlivem nepříznivého počasí.



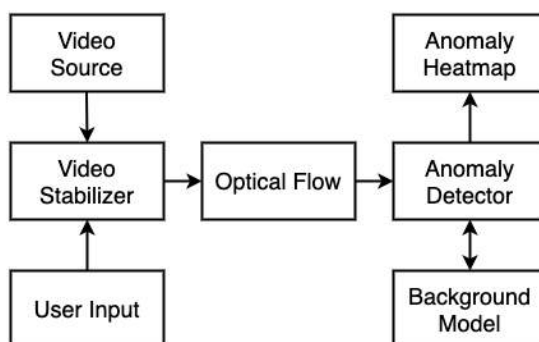
Obrázek 3.1: Ukázka typických videodat z dronu.

Videodata snímaná dronem mají svoje specifika. Jeho stacionární poloha může být narušena náhlým poryvem větru, korekcí trajektorie letu nebo dalšími nežádoucími okolnostmi. Částečně je tento problém řešen pomocí stabilizačního mechanismu úchytu kamery, nicméně i ten má svoje omezení. Je nutné počítat i s možností krátkodobého výpadku v přenosu dat.

Uživatel potřebuje mít přehled o situaci celkového proudění davu osob a potřebuje být upozorněn zejména na jeho zrychlení, zastavení, rozdělení nebo sročení. Změna v proudění davu nemusí být vždy způsobena výskytem nebezpečné situace, ale zároveň představuje její indikátor. Vyhodnocení potenciálního nebezpečí je v odpovědnosti uživatele a i z tohoto důvodu je zde tolerován výskyt falešných upozornění. Při řešení úlohy je nutné vzít v úvahu nedostupnost jakýchkoliv trénovacích dat specifických pro scénu, nad kterou bude analýza prováděna.

V rámci reakce na nebezpečnou událost je nutná intuitivní identifikace místa jejího výskytu. Z tohoto důvodu je klíčové udržovat mapování obrazových dat na mapový podklad. Na základě rozhovorů s potenciálními uživateli byla vyhodnoceno promítání teplotní mapy vyjadřující úrovně anomálie do mapového podkladu jako optimální varianta.

Na základě bližší specifikace úlohy bylo možné sestavit základní schéma komponent uživatelského nástroje, které je ilustrované obrázkem 3.2. Na vstupních videodatech bude vzhledem k jejich specifikaci nutné provádět stabilizaci pro odstínění záchvěvů kamery. Z pohledu korektního promítání výsledků analýzy do mapového podkladu je vhodné zvážit využití uživatelského vstupu. Na úlohu stabilizace videa navazuje detektor anomálie, pro který bude zásadní volba vhodného způsobu extrakce příznaku a modelu pozadí.



Obrázek 3.2: Základní architektura uživatelského nástroje.

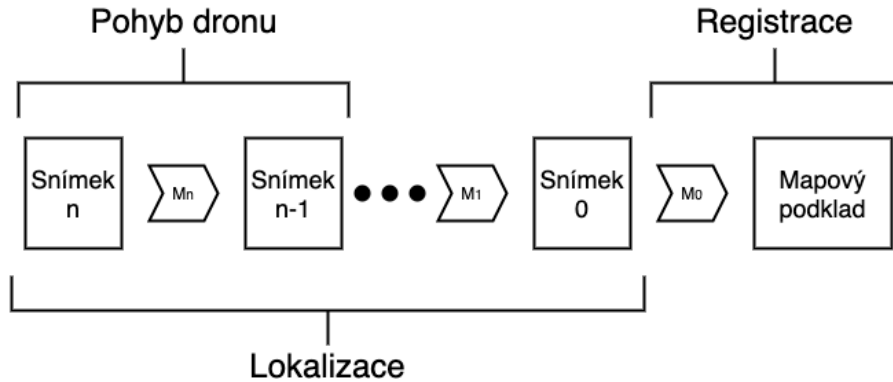
Návrh stabilizátoru videa bude společně s volbou vyhovujícího uživatelského vstupu proveden v kapitole 3.2. Navazující systém pro analýzu chování davu je blíže specifikován v kapitole 3.3.

## 3.2 Stabilizace obrazu

Z popisu charakteru videodat v sekci 3.1 vyplývá nutnost rozlišování mezi pohybem snímacího zařízení a mezi samotným pohybem ve scéně. Vzhledem ke snímání poměrně rozsáhlé oblasti představuje i drobný záchvěv kamery významně rušivý element při zkoumání pohybu v porovnání s pomalým prouděním davu. Stabilizace obrazu je nezbytná i z důvodu udržování kontextu přicházejících videodat vzhledem k mapovému podkladu. Některé drony poskytují GPS údaje společně s dalšími telemetrickými daty. Na základě těchto informací by

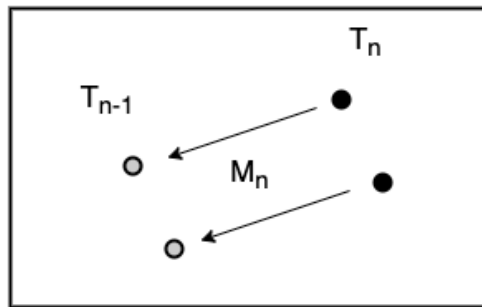
mohla být registrace videodat do mapového podkladu realizovatelná algoritmicky. Nicméně na dostupnost těchto dat se nelze spoléhat a pro získání potřebných informací se tedy nabízí využití uživatelského vstupu.

Přehled transformačních matic použitých při stabilizaci videa a registraci do mapového podkladu je ilustrován obrázkem 3.3. Před začátkem analýzy lze od uživatele manuálně získat odpovídající body ve videodatech a mapovém podkladu a následně vypočítat transformační matici  $M_0$ . K udržení kontextu přicházejících videodat k mapovému podkladu musí být udržována ještě transformační matice mezi počátečním a aktuálním snímkem, kterou lze určit skládáním dílčích transformací mezi jednotlivými snímky. Tyto úvahy čerpají z vlastností transformačních matic blíže specifikovaných v kapitole 2.3.



Obrázek 3.3: Přehled transformačních matic použitých při stabilizaci videa a registraci do mapového podkladu.

Z pohledu stabilizace je nutné řešení rozdělit na stabilizaci drobných záchvěvů a na situaci extrémního pohybu (nebo výpadků přenosu), kdy se může celá scéna posunout mimo záběr kamery. Drobné korekce nestability záznamu mohou být na základě rozboru z kapitoly 2.3 poměrně spolehlivě řešeny detekcí klíčových bodů a sledováním jejich pohybu do následujícího snímku. Řešením této podúlohy je nalezení transformační matice popisující homografii mezi dvěma po sobě jdoucími snímky, což je ilustrováno obrázkem 3.4.



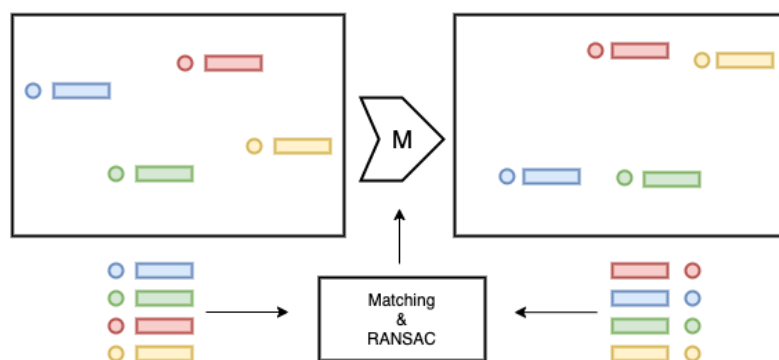
Obrázek 3.4: Princip stabilizace drobných záchvěvů.

Problémem navrhovaného přístupu je pohyb davu nebo jiných objektů ve scéně. V případě detekce klíčových bodů na pohybujících objektech a jejich sledování do následujícího snímku je do homografie započítán pohyb nesouvisející se záchvěvy kamery. To lze vyřešit uživatelským vstupem v podobě binární masky. Pomocí této masky uživatel specifikuje

oblasti, ve kterých není žádný pohyb očekáván a klíčové body z těchto oblastí mohou být bezpečně použity pro určení homografie.

Výpočet homografie blíže diskutovaný v kapitole 2.3 dále uvažuje transformaci pouze mezi dvěma 2D souřadnicovými systémy - souřadnicemi vstupního videa a mapového podkladu. Reálná scéna snímaná dronem je ale 3D charakteru a z tohoto důvodu by při hledání homografie mohlo dojít k dalším nepřesnostem. Vzniká tedy další požadavek na binární masku zadávanou uživatelem - měla by zahrnovat pouze plochy ležící ve stejné rovině.

V případě silného záchvěvu nebo úplné ztráty původní scény není možné využít předchozí přístup s dostatečnou přesností. Namísto toho lze ve scéně na základě informací z kapitoly 2.3 pravidelně detekovat vhodné klíčové body společně s jejich deskriptory, což je ilustrováno na obrázku 3.5. Následným hledáním odpovídajících klíčových bodů mezi původním a aktuálním snímkem lze určit hledanou homografii. Tento přístup je více robustní, ale zároveň i výpočetně náročnější.



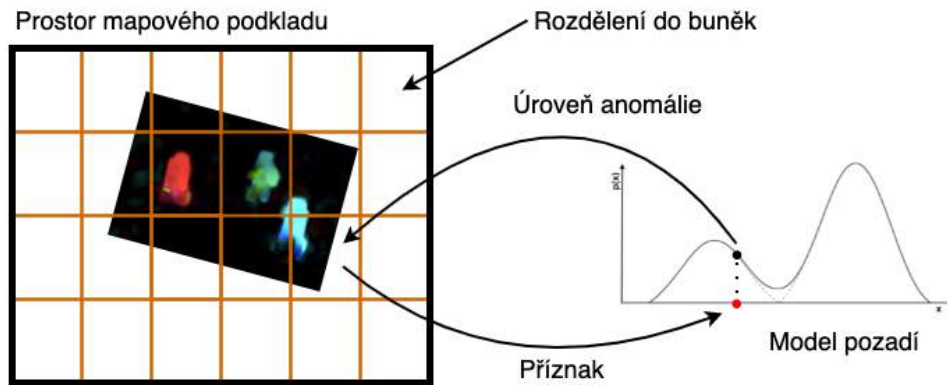
Obrázek 3.5: Princip stabilizace extrémních záchvěvů a zpětného dohledání scény.

### 3.3 Systém detekce anomálie

Na základě průzkumu existujících přístupů z kapitoly 2.1 byl vytvořen návrh základní struktury detektoru anomálie ilustrovaný na obrázku 3.6. Častým způsobem detekce anomálie v chování davu je zkoumání změn optického toku [41, 19, 29]. Existuje mnoho metod pro jeho výpočet, nicméně nutnost zpracování v reálném čase částečně limituje možnost využití některé z metod založené na konvolučních neuronových sítích.

Vypočtený optický tok je vzhledem k zachování prostorového kontextu nejprve transformován do prostoru mapového podkladu za použití transformačních matic získaných při řešení úlohy stabilizace obrazu. Prostor mapového podkladu je rozdělen do mřížky tvořené buňkami o souhlasné velikosti. Základním principem výpočtu úrovně anomálie pro danou buňku je modelování optického toku v daném místě. Porovnáním aktuálního optického toku s jeho modelem je pomocí vhodné metriky vypočtena hodnota anomálie.

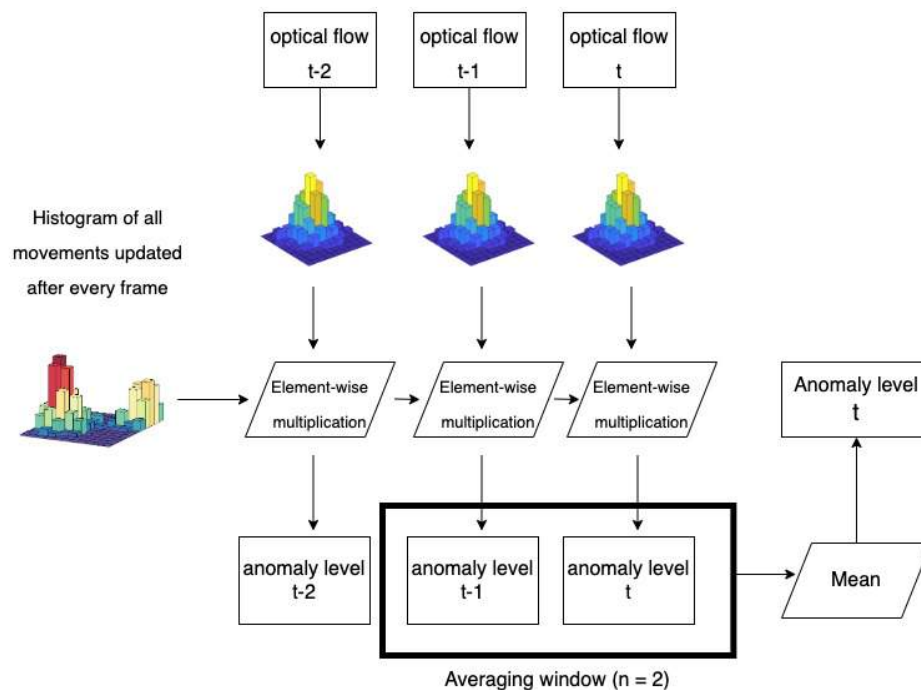
Specifikem řešené úlohy je nedostupnost jakýchkoliv trénovacích dat pro scénu, ve které bude detekce anomálie prováděna. Metoda detekce anomálie tedy musí být schopna vytvářet a průběžně aktualizovat model pohybu ve scéně pouze na základě on-line dat přicházejících z dronu, a to bez dostupnosti anotací. V rámci této práce byly navrženy dvě metody detekce anomálie, které se liší ve způsobu vytváření modelu pozadí i v následném výpočtu hodnoty anomálie.



Obrázek 3.6: Funkční součásti detektoru anomálie.

### 3.4 Metoda založená na histogramech

Statistická metoda založená na analýze histogramů optického toku uchovává model obvyklého pohybu v podobě normalizovaného 2D histogramu. Pixely optického toku náležící dané buňce jsou podle rychlosti a směru pohybu rozděleny do vhodného počtu skupin a hodnota histogramu pro danou kombinaci rychlosti a směru představuje četnost jejího výskytu.



Obrázek 3.7: Metrika pro výpočet úrovně anomálie na základě porovnání histogramů optického toku.

Porovnáním histogramu optického toku v aktuálním snímku se souhrnným histogramem optického toku pro určitý počet předcházejících snímků lze určit hodnotu odchylky od běžného pohybu pro danou buňku. Zásadní vliv na úspěšnost této metody představuje

bezpochyby metrika pro porovnání podobnosti dvojice histogramů. Použitá metrika musí zohledňovat možnost občasného výskytu šumu ve výpočtu optického toku, která může být způsobena chybou metody pro jeho výpočet nebo nepřesností stabilizace obrazu. Respektován musí být i současný výskyt více pohybu v jedné buňce současně.

Vhodnou metrikou pro porovnání dvojice histogramů je jejich násobení po prvcích a následný součet prvků vzniklé matice, což je vyjádřeno vztahem 3.1. Výsledek této operace spadá do intervalu  $< 0; 1 >$  podle úrovně podobnosti (v případě normalizovaných histogramů na vstupu). Druhou metrikou často používanou pro porovnání dvojice histogramů je výpočet součtu jejich průniku [5], což lze formálněji zapsat vzorcem 3.2.

$$sim(x, y) = \sum_{i=1}^d x_i y_i \quad (3.1)$$

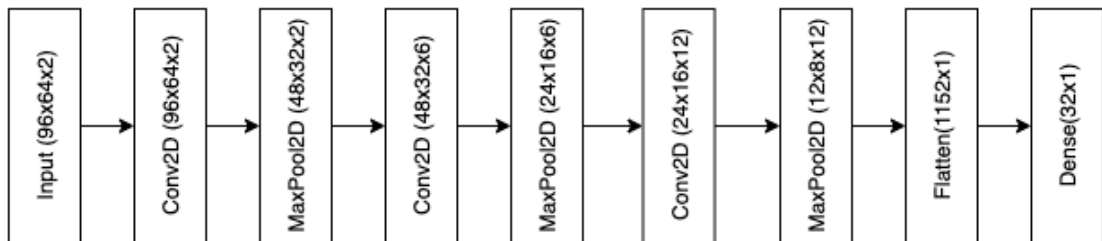
$$sim(x, y) = \sum_{i=1}^d \min(x_i, y_i) \quad (3.2)$$

Problém nestability výsledku metody optického toku byl vyřešen ukládáním několika posledních výsledků porovnání. Úroveň anomálie je poté aritmetickým průměrem několika posledních hodnot. Diagram popisující navrhovaný způsob výpočtu úrovně anomálie je na obrázku 3.7.

### 3.5 Metoda založená na extrakci příznaku

Hlavní myšlenka druhé metody pro detekci anomálie spočívá v popisu optického toku dané buňky pomocí relevantního příznakového vektoru [41]. Na základě série příznakových vektorů je průběžně aktualizován generativní model jejich pravděpodobnostního rozložení. Úroveň anomálie dané buňky je následně určena porovnáním aktuálního příznaku s modelem.

Pro extrakci příznakového vektoru bylo využito neuronové sítě s architekturou typu *Denoising Convolutional Autoencoder* [41]. Tato neuronová síť se rozděluje na část kodéru a dekodéru. Kodér nejprve provádí kompresi vstupního optického toku dané buňky do příznakového vektoru o výrazně nižší dimenzionalitě oproti vstupním datům. Na základě tohoto vektoru usiluje symetricky odpovídající dekodér o rekonstrukci původního vstupu. V průběhu trénování je tedy identifikována klíčová informace, která musí být do příznaku zakódována pro úspěšnou rekonstrukci původního vstupu.



Obrázek 3.8: Navrhovaná architektura neuronové sítě pro extrakci příznakového vektoru z výřezu optického toku.



Schéma navrhované architektury neuronové sítě inspirované článkem [41] je na obrázku 3.8. Vstupem neuronové sítě je výřez optického toku, který náleží dané buňce modelu. Následuje postupné snižování rozlišení a zvyšování počtu kanálů vstupního vzorku obvyklým způsobem pomocí konvolučních a seskupujících vrstev. V poslední fázi je tato informace zakódována pomocí plně propojené vrstvy pouze do nízkodimenzionálního příznaku, pomocí kterého lze na on-line datech vytvářet model pozadí.

Úspěšnost navrhované architektury neuronové sítě lze dále optimalizovat změnou domény vstupních dat. Anomálie v davu nemusí být vždy patrná pouze z optického toku. Z pohledu řešení úlohy považuji za vhodné vyhodnotit vliv využití obrazové informace nebo optického toku několika posledních snímků.

Vhodným a velmi často používaným přístupem pro modelování pravděpodobnostního rozložení multidimenzionálních dat je využití směsi Gaussovských rozložení [31]. Nicméně průzkum existujících implementací poukázal na nedostupnost vhodného knihovního řešení s možností inkrementálního trénování na online datech. Z tohoto důvodu jsem se rozhodl navrhnout vlastní implementaci.

Navrhovaný algoritmus pracuje s množinou Gaussovských rozložení, které jsou definovány počtem prvků, lineárním součtem a kvadratickým součtem. Ve výchozím stavu je tato množina prázdná. Algoritmus umožňuje průběžnou aktualizaci modelu pozadí pomocí příznakového vektoru. V prvním kroku aktualizace je nalezena vzdálenost příznakového vektoru a středu nejbližší Gaussovy křivky z dané množiny. Pokud je tato vzdálenost menší než stanovený práh, dojde k aktualizaci statistik nejbližší Gaussovy křivky. Pokud je množina Gaussových rozložení prázdná nebo pokud vzdálenost překračuje stanovený práh, je inicializována nová Gaussova křivka se středem odpovídajícím danému příznaku.

Na operaci aktualizace modelu navazuje výpočet pravděpodobnosti výskytu daného příznakového vektoru vzhledem k modelu pozadí. V první fázi je pro každou Gaussovu křivku vypočtena úroveň odpovědnosti za jeho vygenerování podle vztahu 3.3. Proměnná  $w_z$  reprezentuje odpovědnost Gaussovy křivky s identifikátorem  $z$  za vygenerování příznaku  $x$ . Proměnné  $\mu$  a  $\sigma^2$  reprezentují parametry GMM modelu.

$$w_z = \frac{\mathcal{N}(x \mid \mu_z, \sigma_z^2)}{\sum_k \mathcal{N}(x \mid \mu_k, \sigma_k^2)} \quad (3.3)$$

Každá Gaussova křivka reprezentuje určitý cluster pohybu davu. Podobnost příznakového vektoru s tímto clusterem je vyjádřena pomocí váhy odpovědnosti  $w_z$ . Kromě podobnosti je zde ale důležité započítat i pravděpodobnost výskytu daného clusteru  $\pi_z$ . Pravděpodobnost příznakového vektoru vzhledem k modelu pozadí je tedy vypočtena podle vztahu 3.4.

$$\begin{aligned} \pi_z &= \text{count}(z) / \sum_k \text{count}(k) \\ p(x \mid \eta) &= \sum_z w_z \pi_z \end{aligned} \quad (3.4)$$



## Kapitola 4

# Experimenty a vyhodnocení

V rámci této práce byly navrženy metody detekce anomálie i stabilizace obrazu. Každá z těchto metod disponuje několika parametry, jejichž optimální hodnotu je nutné určit pomocí relevantního vyhodnocení. Z pohledu praktického nasazení je potom důležité identifikovat podmínky, za kterých je dosaženo správných výsledků a za kterých naopak metody selhávají. Cílem této kapitoly je návrh vhodných experimentů pro ověření funkce vyvíjených metod, jejich vyhodnocení pomocí standardních metrik a následné zhodnocení výsledků.

### 4.1 Datové sady

Pro vývoj metod strojového učení obecně je zcela zásadní dostupnost relevantních datových sad pro trénování a následné vyhodnocení jejich úspěšnosti. Problémem úloh detekce anomálie je přílišná subjektivnost specifikace daného problému. Různé metody definují pojem anomálie odlišným způsobem, což komplikuje jejich relevantní srovnávání [20]. V následující kapitole bude proveden průzkum existujících datových sad s cílem identifikace vhodné datové sady relevantní k uživatelské úloze blíže specifikované v kapitole 3.1.

Metody detekce anomálie v datových scénách lze rozdělit do dvou základních skupin, od kterých se dále odvíjí podoba trénovacích dat a jejich anotací [19]. Část metod je trénována na datech s výskytem běžného i abnormálního chování. K těmto datům jsou připojeny anotace o výskytu anomálie, a to globálního nebo lokálního charakteru. Druhý přístup předpokládá výskyt pouze běžného chování v trénovacích datech a metody jsou trénovány bez přítomnosti učitele.

### UCSD Dataset

Datová sada UCSD Dataset [7] je tvořena krátkými videosekvencemi o rozlišení 238x158 pixelů skládajících se z 200 snímků. Byla pořízena pomocí stacionární kamery snímající chodník z vyvýšené pozice. Hustota osob na záznamech kolísá od silně zalidněných scén po výskyt pouhých několika chodců. Dataset je rozdělený do dvou nezávislých částí, které odpovídají různým scénám. Ukázka dat je na obrázcích 4.1.

Pro každou scénu byla pořízena trénovací i testovací podčást. Trénovací podčást obsahuje záznamy pouze bez výskytu anomálií. Anomálie je v rámci tohoto datasetu definována jako výskyt nežádoucích objektů (cyklista, skateboardista, vozičkář nebo vůz) nebo neobvyklý vzor pohybu. K veškerým trénovacím snímkům je dostupná anotace o globálním výskytu anomálie. K několika vybraným sekvencím jsou dále dostupné masky definující



Obrázek 4.1: Ukázka dat z UCSD datasetu. (Převzato z [7])

lokalitu anomálie. Podrobnější informace o počtech videosekvencí tohoto datasetu jsou uvedeny v tabulce 4.1.

Název scény	Trénovací sekvence	Testovací sekvence	Počet sekvencí s globální anotací
Peds1	34	36	10
Peds2	16	12	12

Tabulka 4.1: Podrobnější informace o rozložení počtu videosekvencí v rámci UCSD datasetu.

Definice neobvyklých situací používaná v tomto datasetu poměrně dobře odpovídá řešené úloze. Neobvyklé objekty se ve většině případů projevují i netypickým pohybovým vzorem. Relevantnost dat podtrhuje i vyvýšené umístění kamery odpovídající možné podobě videodat z dronu. Z těchto důvodů považuji tento dataset vyhovující pro vyhodnocení navrhovaných metod.

## Avenue Dataset

Tento dataset [22] tvoří 16 trénovacích a 21 testovacích videosekvencí o celkovém počtu 30652 snímků s rozlišením. Trénovací data zahrnují pouze situace s obvyklým chováním a anomálie se vyskytují pouze v testovacích datech. Neobvyklé situace jsou v anotacích označeny pomocí ohraničujících rámečků.



Obrázek 4.2: Ukázka dat z Avenue datasetu. (Převzato z [22])

Videa byla pořízena na veřejném prostranství s kamerou ve stacionární poloze. Občasné dochází k záchvěvům kamery nebo řídkému výskytu některého z druhů běžného chování. Anomálii v testovacích datech představuje celkem 14 typů neobvyklých situací (běh, házení objektů, nezvykle pomalá chůze a další). Ukázka dat tohoto datasetu je na obrázku 4.2. Videodata jsou velmi odlišná od charakteru záznamů pořízených dronem, nicméně výskyt netypických pohybových vzorů může velmi dobře posloužit pro vyhodnocení obecnosti detektoru anomálie.

## UCF Crowd Dataset

Tento dataset [1] se skládá z celkem 38 krátkých videosekvencí s délkou od několika vteřin do jedné minuty. Jejich rozlišení se pohybuje od 480x360 pixelů do 720x480 pixelů. Jedná se o poměrně rozmanité scénérie snímáné statickou kamerou. Scénérie zahrnují různorodé dopravní situace, davové scény, ale i abstraktní pohybující se grafické vzory. Tento dataset považuji za přínosný zejména pro trénování metody extrahující příznakový vektor popisující pohyb, a to bez přítomnosti učitele. Ukázka dat je na obrázku 4.3.



Obrázek 4.3: Ukázka dat z UCF datasetu. (Převzato z [1])

## UMN Crowd Dataset

Jedná se o další veřejně dostupnou datovou sadu [28] zaměřenou na detekci anomálie v chování davu. Celkově se skládá z 11-ti videosekvencí o celkové délce 4 minuty a 17 vteřin. Scéna je vždy snímána pomocí stacionární kamery umístěné na vyvýšené pozici. Každá videosekvence se rozděluje na části s obvyklým a netypickým chováním. Netypické chování zde obvykle představuje chaos v rámci celé scény. Ukázka dat je na obrázcích 4.4. Tato datová sada velmi dobře odpovídá řešené úloze a vzhledem k její jednoduchosti ji považuji za výchozí bod experimentů.



Obrázek 4.4: Ukázka dat z UMN datasetu. (Převzato z [28])

## Train Station Dataset

Datová sada [43] tvořená 33 minut dlouhým záznamem nádražní haly o rozlišení 720x480 pixelů. Záznam je pořízený z vyvýšeného místa pomocí stacionární kamery. Po celou dobu záznamu se zde vyskytuje poměrně velké množství osob vztažených k danému prostoru. Anotace obsahují trajektorie pohybu chodců ve scéně, což umožňuje provádění analýzy davového chování. Ukázka dat je na obrázku 4.5. Tento videozáznam by mohl být vzhledem k podobnosti dat se záznamy pořízenými dronem využit zejména pro trénování metody pro extrakci relevantního příznakového vektoru bez přítomnosti učitele.



Obrázek 4.5: Ukázka dat z Train Station Datasetu. (Převzato z [43])

## Multidrone Public Dataset

Z ostatních datových sad stojí za zmínku datová sada Multidrone Public Dataset [23, 24]. Je to rozsáhlá datová sada sestávající z několika desítek hodin různorodých záznamů s výskytem scén lidských davů snímáných dronem. Datová sada je v HD rozlišení a pro akademické účely je volně dostupná na vyžádání. Vzhledem k nedostupnosti anotací o výskytu anomálie je vhodná zejména pro trénování metod bez učitele. Ukázka relevantních dat je na obrázku 4.6.



Obrázek 4.6: Ukázka ze záznamu přeletu dronu nad veřejným prostranstvím pocházející z datové sady Multidrone Public Dataset (Převzato z [23, 24]).



## Pořízení vlastního datasetu

V rámci předcházejícího průzkumu nebyla identifikována žádná datová sada obsahující záznamy davových scén snímáných dronem. Dostupnost těchto dat může být zásadní zejména z pohledu relevantního vyhodnocení stabilizační metody. Z tohoto důvodu byla v rámci této práce pořízena i vlastní videodata.



Obrázek 4.7: Ukázka dat z vlastního záznamu sportovního hřiště.

Vzhledem k uživatelské úloze považuji za velmi přínosný záznam sportovního hřiště o celkové délce 213 vteřin a rozlišení 3840x2160 pixelů. Hřiště bylo snímáno pomocí dronu z výšky v rozmezí 60 - 80 metrů. Po obvodu hřiště se pohybuje skupina 23 osob podle nacvičeného scénáře. Scénář zahrnuje náhlé zastavení, změnu směru pohybu, chaotický útěk a simulovaný silný záchvěv dronu. K záznamu nejsou připraveny žádné anotace, nicméně jej lze využít v rámci ověření správného postupu ve vývoji metod detekce anomálie. Ukázka záznamu je na obrázku 4.7.



Obrázek 4.8: Ukázka dat z vlastních záznamů pořízených stacionární kamerou.

Další sada 3 videozáznamů byla pořízena stacionární kamerou z budovy ve výšce přibližně 10 metrů nad terénem. Video byla nahrávána v rozlišení 1920x1080 pixelů s celkovou délkou přibližně 12 minut. Na záznamech se pohybuje osoba podle předem definovaného scénáře. Kromě obvyklých pohybů se zde vyskytují anomálie v podobě nečekané změny směru a náhlého zastavení nebo zrychlení. Ukázka záznamů je na obrázcích 4.8.

Poslední skupina 4 videí o rozlišení 1920x1080 pixelů a celkové délce 21 minut byla pořízena na dopravním odpočívadle poblíž málo frekventované komunikace. Skupina 5 chodců zde realizuje několik průchodů na konec scény a zpět. V rámci těchto průchodů je zde podle scénáře sehraána potyčka končící ležící osobou na zemi. Dále je zde realizována anomálie v podobě nečekaného přechodu z chůze v běh nebo průjezd osobního vozu. Videodata zahrnují věrohodné pohyby dronu menšího i rozsáhlejšího charakteru, čehož může být využito například pro vyhodnocení funkce stabilizační metody. Ukázka ze záznamu je na obrázku 4.9.



Obrázek 4.9: Ukázka dat z vlastního záznamu z okolí dopravního odpočívadla

## 4.2 Vyhodnocení metody stabilizace

V posledních letech bylo navrženo mnoho metod stabilizace videa. Nicméně univerzální metrika pro jejich srovnání stále nebyla přesně specifikována. Problémem v porovnávání těchto metod je nutnost jejich vyhodnocení z více různých hledisek [14]. Metriky vyhodnocení lze rozdělit na subjektivní a objektivní.

Mezi objektivní metriku pro hodnocení metody stabilizace videa patří ztráta rozlišení způsobená stabilizací [14]. Metoda použitá pro řešení této úlohy je nicméně poměrně specifická a jejím cílem je zejména nalezení transformace mezi jednotlivými snímky. Z tohoto důvodu je tento způsob vyhodnocení nevyhovující.

Mezi další objektivní metriky patří PSNR (*peak signal to noise ratio*) nebo SSIM (*structural similarity*) [14]. Tyto metriky slouží k vyhodnocení úrovně podobnosti mezi dvojicí sousedních obrázků. Pro vyhodnocení navrhované stabilizační metody bylo využito metriky MDE (*Mean displacement error*) [18] z důvodu její souvislosti z řešenou úlohou. Tato metrika hodnotí přímo kvalitu nalezené transformace a vyjadřuje průměrnou odchylku v posunu jednotlivých pixelů oproti anotaci.

Pro vyhodnocení metody stabilizace byla vytvořena vzhledem k její specifičnosti vlastní datová sada. Tato datová sada se skládá z vlastních videí pořízených kamerou umístěnou na palubě dronu společně se záznamy veřejného prostranství z datové sady Multidrone Public Dataset [23, 24]. Celkem se jedná o 9 videosekvencí opatřených manuální anotací s maskou definující stabilní oblasti ve videu a transformační maticí specifikovanou člověkem.

Číslo videa	Popis	MDE (px)	MDE (px) bez stabilizace
01	Dron ve statické poloze, prudký záchvív se ztrátou původní scény.	4	6
02	Drobné záchvěvy dronu, ztráta původní scény na 7 vteřin	3	8
03	Přelet konstantním směrem a rychlostí. Lehký pohyb i ve stabilních oblastech.	8	223
04	Přelet konstantním směrem a rychlostí.	8	200
05	Přelet konstantním směrem a rychlostí.	4	139
06	Plynulé ale výrazné natočení kamery od scény a zpět. Část scény vždy viditelná.	6	116
07	Plynulé ale výrazné natočení kamery od scény a zpět. Část scény vždy viditelná.	9	118
08	Plynulý posun konstantní rychlostí.	7	242
09	Mnoho drobných záchvěvů, velký záchvív se ztrátou scény, málo stabilních oblastí.	6	28

Tabulka 4.2: Výsledky vyhodnocení navrhované metody stabilizace videa na vlastní datové sadě.

Výsledky vyhodnocení jsou shrnuty v tabulce 4.2. Je zde uvedeno porovnání průměrné odchylky jednotlivých pixelů bez použití a se zapnutou stabilizační metodou. První dvě videa demonstrují úspěšné dohledání původní scény po její kompletní ztrátě vlivem prudkého záchvěvu. Se zapnutou stabilizací obrazu došlo u ostatních videí k výraznému poklesu odchylky, která je vzhledem k rozlišení vstupního videa 640x360 pixelů prakticky zanedbatelná. Bylo tedy prokázáno, že stabilizační metoda uspokojivě vyřešila veškeré scénáře testovací datové sady.

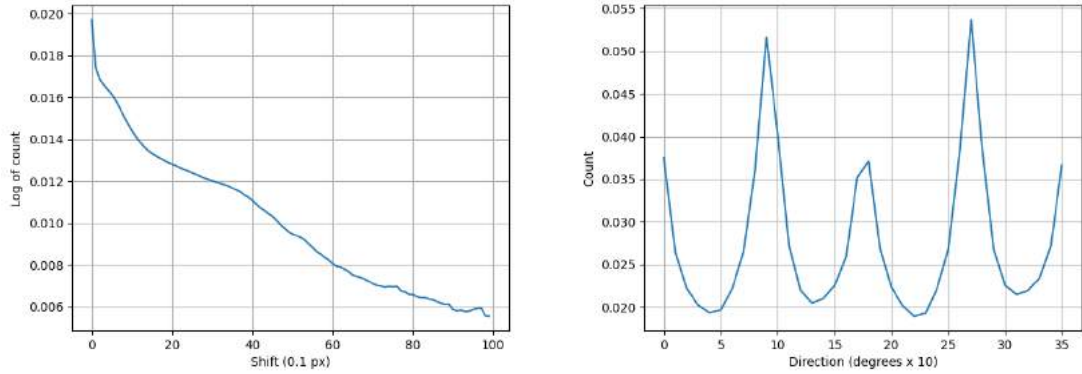
### 4.3 Experimenty s detektory anomálie

Cílem této kapitoly je příprava detektorů anomálie navržených v kapitole 3.3 na vyhodnocení reálnými daty. V rámci přípravy metody založené na histogramech optického toku bylo nutné provést návrh relevantních experimentů vedoucích k volbě jejího nastavení. U metody využívající konvoluční autoenkodér bylo zásadní zamyslet se nad výchozí architekturou a navrhnout její modifikace, které mohou potenciálně vést ke zlepšení úspěšnosti detekce. Vzhledem k optimalizaci výpočetních nároků i úspěšnosti detekce bylo pro obě metody zásadní provést volbu vhodného rozlišení modelu pozadí.

### Volba parametrů metody založené na histogramech

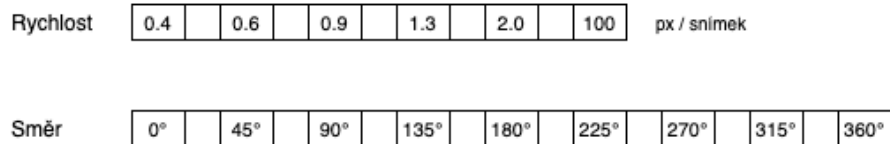
Detektor anomálie založený na histogramech optického toku z kapitoly 3.3 představuje základní a intuitivní přístup, jakým lze k detekci anomálie přistupovat. Sémantika histogramu pohybu je na rozdíl od přístupů extrakce příznakového vektoru pomocí neuronových sítí předem definována, což umožňuje jeho srozumitelnou reprezentaci pomocí sloupcového grafu.

Důležitým parametrem ovlivňujícím úspěšnost detekce anomálie je v tomto případě volba rozsahu a rozlišení samotného histogramu. Pro volbu tohoto parametru bylo nezbytné zjistit rozložení vzorků optického toku vzhledem ke směru a rychlosti pohybu. Měření bylo provedeno na datové sadě UMN dataset [28] vzhledem k její dobré podobnosti s daty cílové domény. Naměřené hodnoty rozložení jsou vizualizovány pomocí grafů na obrázcích 4.10.



Obrázek 4.10: Rozložení vzorků optického toku podle rychlosti a směru pohybu naměřené v datové sadě UMN dataset [28]

U směru pohybu lze pro obecnou scénu uvažovat jeho rovnoměrné rozložení v intervalu  $\langle 0; 2\pi \rangle$ . Rozložení rychlostí pohybu muselo být do grafu zakresleno v logaritmické doméně z důvodu vysokých hodnot pro pohyby drobného charakteru. Vzhledem k charakteru rozložení dat je tedy vhodné provést mapování vzorků rychlosti na třídy histogramu pomocí nelineární funkce.

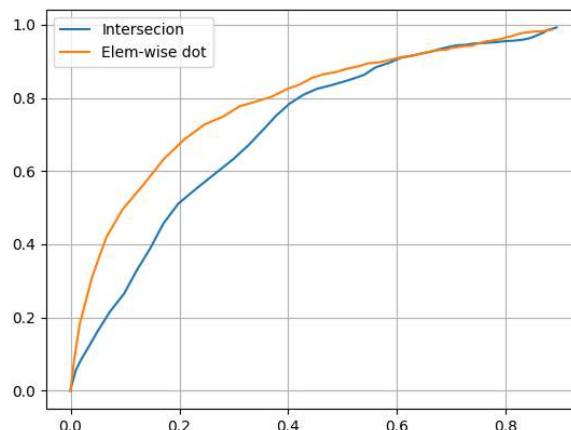


Obrázek 4.11: Volba intervalů 2D histogramu optického toku.

Vzorky s příliš nízkou rychlostí pohybu nemají z pohledu analýzy příliš dobrou vypovídající hodnotu a je vhodné provést jejich filtraci. Během prvotních spuštění metody na videích vlastní datové sady byl rovněž identifikován problém s výraznou nestabilitou směru pohybu těchto vzorků. Podobně je tomu se vzorky s vysokou rychlostí, které se v datasetu vyskytují pouze ve velmi malé míře a pro analýzu nepřinášejí užitečnou informaci. Při volbě rozlišení histogramu je nutné dbát na výpočetní náročnost výpočtu i následného porovnávání histogramů. Výsledná volba intervalů použitých v 2D histogramu je ilustrována obrázkem 4.11.

Vliv na úspěšnost statistické metody má rovněž metrika porovnání dvojice histogramů. Z tohoto důvodu bylo provedeno srovnání navrhovaných metod z kapitoly 3.3 na UMN datasetu [28]. Z výsledků vyhodnocení na obrázku 4.12 je patrné, že o něco lepších výsledků dosahuje varianta realizující násobení histogramů po prvcích oproti výpočtu jejich průniku.



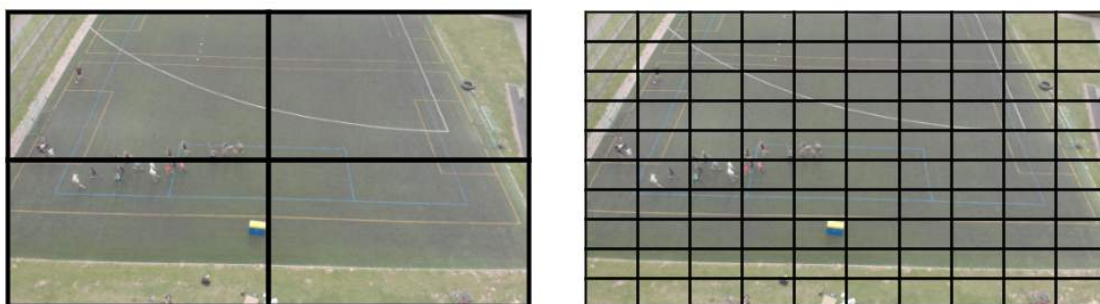


Obrázek 4.12: Porovnání metrik srovnávání histogramů na datové sadě UMN dataset.

Dalším důležitým parametrem navrhovaného statistického přístupu je volba velikosti plovoucího okna, která slouží k odstranění šumu z detekcí. Příliš malé plovoucí okno může mít za následek vyšší počet falešných detekcí. V opačném případě zde hrozí potlačení upozornění na skutečnou anomálii. Na videích z reálných scén se ukázala velikost okna mezi 3 a 5 vzorky jako dostačující v rámci potlačení šumu.

## Optimalizace rozlišení mřížky

Jedním z klíčových faktorů ovlivňujících úspěšnost detektorů anomálie je rozlišení mřížky, do které je prostor modelu pozadí rozdělen. Vliv rozlišení mřížky na prostorovou souvislost mezi optickým tokem a modelem pozadí ilustruje situace na dvojici obrázků 4.13.



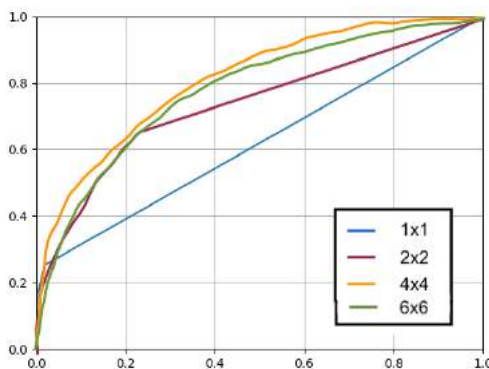
Obrázek 4.13: Ilustrace vlivu rozlišení mřížky rozdělovací model pohybu na výsledky metody detekce.

Na obrázku vlevo je použitá mřížka příliš hrubá. Pro veškeré buňky modelu jsou v tomto příkladě obvyklé pouze oblasti bez pohybu. Při průchodu osob danou buňkou je jejich pohyb ve srovnání se zbytkem příliš malý a anomálie zde prakticky nemůže být detekována. Navíc je pro tento model lhostejné, zda se osoby pohybují prostředkem herní plochy, nebo po úplném okraji.

Obrázkem vpravo je ilustrován opačný problém. Tento model pozadí je příliš citlivý na lokalitu pohybu z důvodu jemné volby mřížky. V případě posunu pohybu snímaných osob v rámci jednotek metrů zde může být nesprávně identifikována vysoká úroveň anomálie.

Cílem tohoto experimentu je tedy odvození optimálních rozměrů buňky modelu vzhledem k reálnému prostoru. Při praktickém nasazení lze poté vhodné rozlišení mřížky vypočítat na základě uživatelem zadané transformace s mapovým podkladem. Při volbě rozlišení nesmí být opomíjen výrazný vliv tohoto parametru na výpočetní nároky.

Metoda detekce založená na histogramech optického toku byla za tímto účelem vyhodnocena na UMN datasetu [28] pro rozlišení mřížky modelu 1x1, 2x2, 4x4 a 6x6 buněk. Pro každou z 11-ti videosekvencí tohoto datasetu byl inicializován prázdný model pozadí a detekce anomálie byla prováděna v on-line režimu. Vzhledem k dostupnosti anotací pouze globálního charakteru byla z úrovně anomálie pro jednotlivé buňky vypočtena maximální hodnota. Porovnání této hodnoty s prahem daným ROC křivkou [11] bylo provedeno rozhodnutí o globálním výskytu anomálie. Výsledky vyhodnocení jsou vyjádřeny pomocí ROC křivky na obrázku 4.14.



Obrázek 4.14: Vyhodnocení statistické metody na UMN datasetu pro různá rozlišení mřížky modelu.

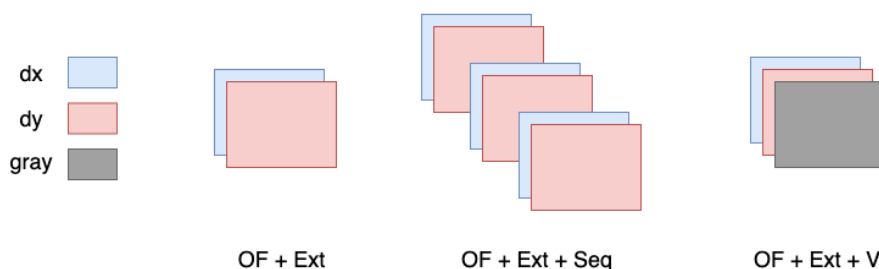
Z výsledků analýzy je patrné, že k největšímu zlepšení detektoru dochází při zvětšování rozlišení mřížky modelu po úroveň 4x4 buňky. Varianty 1x1 a 2x2 vyhodnocují většinu snímků nulovou anomálií, což nelze považovat za uspokojivý výsledek. Další zjemňování mřížky již nepřináší příliš velká zlepšení, ale kvadraticky navyšuje složitost výpočtu. Z vizuální kontroly videosekvencí datasetu UMN lze usoudit, že rozměry snímaných scén činí přibližně 30x30 metrů. Hrubým odhadem lze tedy prohlásit, že ideální rozměry hrany jediné buňky modelu jsou okolo 8 metrů.

## Trénování autoenkodéru

Původní konvoluční autoenkodér navržený v kapitole 3.3 posloužil jako výchozí bod veškerých experimentů. Na jeho základě bylo navrženo několik dalších variant a ty byly poté vyhodnoceny na datových sadách UCSD [7], UMN [28] a Avenue [22]. Úspěšnost modelů byla ověřena v **off-line** i **on-line** režimu a bylo provedeno porovnání jejich úspěšnosti s existujícími přístupy.

Výchozí architektura konvolučního autoenkodéru (varianta **OF**) byla dále doplněna o další plně propojenou vrstvu o celkovém počtu 1152 neuronů (varianta **OF + Ext**). Se vzrůstajícím počtem parametrů modelu se zde otevírá možnost zachycení komplexnější souvislosti ve vstupních datech. Poslední navrhovanou variantou založenou pouze na analýze optického toku byla architektura, jejíž vstup tvoří trojice sousedních snímků optického toku (varianta **OF + Ext + Seq**). Příznakový vektor by mohl být v tomto případě robustnější vůči nestabilitě vstupního optického toku a zároveň by zde mohla být zachycena i změna pohybu.

V některých situacích nemusí být výskyt anomálie patrný pouze na základě optického toku, ale důležitou roli zde může sehrát i vizuální informace. Například v datové sadě UCSD je za jednu z anomálií považován průjezd skateboardisty. Nicméně skateboardista se od běžného chodce liší pouze svým vzhledem a v doméně optického toku je při zachování souhlasné rychlosti a směru pohybu nerozpoznatelný. Poslední návrh konvolučního autoenkodéru (varianta **OF + Ext + V**) má tedy na vstupu kromě jednoho vzorku optického toku i obrazová data v odstínech šedé. Přehled veškerých modifikací domény vstupních dat je uveden na obrázku 4.15.



Obrázek 4.15: Přehled zkoumaných modifikací domény vstupních dat konvolučního autoenkodéru.

Pro účely trénování byla nejprve využita datová sada Train Station Dataset [43]. V této datové sadě se vyskytuje pohyb mnoha osob snímáný z kamery umístěné vysoko nad scénou, což velmi dobře odpovídá situacím řešené úlohy. Tato datová sada byla později rozšířena o videosekvence datové sady UMN, UCF [1] a UCSD, aby mohl být zkoumán vliv rozmanitějších dat na úspěšnost detekce. V tomto případě byla ale trénovací data natolik rozmanitá, že se nedařilo rozumným způsobem minimalizovat chybovou funkci.

Ukázalo se, že zcela zásadní vliv na výsledky trénování má volba aktivačních funkcí. Pro aktivaci plně propojených vrstev byla zvolena aktivační funkce sigmoid vzhledem ke vhodnosti normalizace hodnot příznakového vektoru do intervalu  $< 0; 1 >$ . Aktivace výstupní vrstvy dekodéru byla ponechána lineární, protože se v rekonstruovaném optickém toku mohou vyskytovat kladné i záporné hodnoty z teoreticky neomezeného rozsahu. U ostatních konvolučních vrstev byla po několika neúspěšných experimentech ponechána rovněž lineární aktivace.

## 4.4 Vyhodnocení na veřejných datasetech

Obvyklou metrikou pro porovnání úspěšnosti detektorů anomálie je vyhodnocení pomocí ROC křivky [11]. Detektor anomálie může být zároveň považován za binární klasifikátor. Pokud je klasifikátoru předložen vzorek dat, nastává vždy jedna ze čtyř variant uvedených na obrázku 4.16. V případě pozitivního výstupu klasifikátoru se na základě porovnání

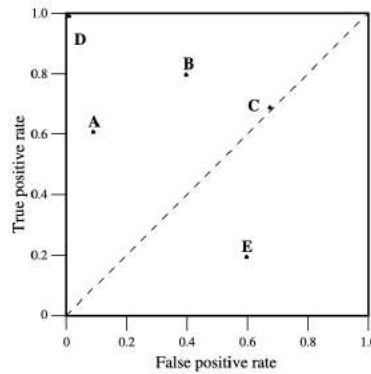
s anotací může jednat o korektně pozitivní (*true positive*) nebo nesprávně pozitivní (*false positive*) výsledek. V případě negativního výstupu klasifikátoru je výsledek kategorizován analogicky.

		<u>True class</u>	
		<b>p</b>	<b>n</b>
<u>Hypothesized class</u>	<b>Y</b>	True Positives	False Positives
	<b>N</b>	False Negatives	True Negatives

Obrázek 4.16: Varianty výstupu binárního klasifikátoru. (Převzato z [11])

ROC graf definuje 2D prostor, ve kterém je metrika *true positive rate* vynášena na osu y a metrika *false positive rate* na osu x. Použité metriky lze určit dosazením do vztahu 4.1. Obvyklým výstupem obecného binárního klasifikátoru je hodnota skóre definující pravděpodobnost správnosti klasifikace. Každé fixní nastavení tohoto prahu odpovídá jednomu bodu ROC křivky. Na základě postupných změn prahu klasifikace lze tedy sestavit kompletní ROC křivku.

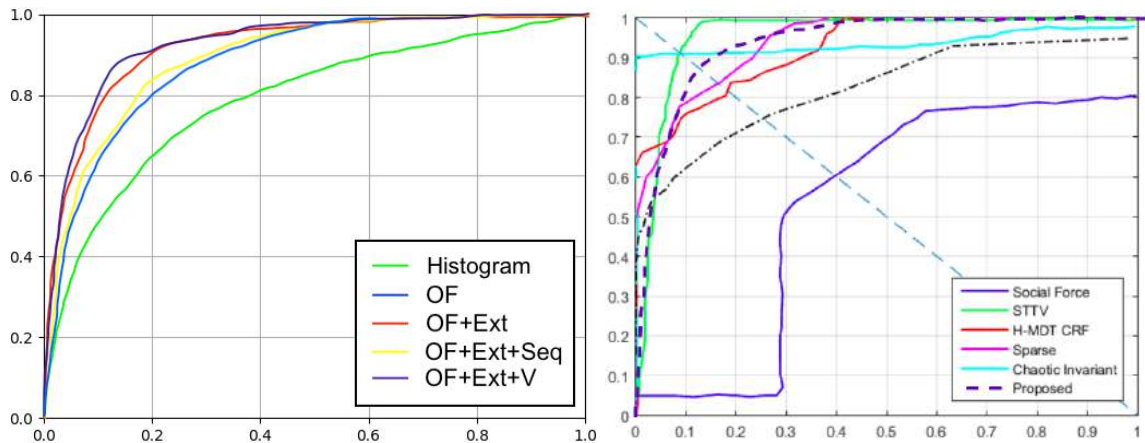
$$\text{false positive rate} = \frac{FP}{N} \quad \text{true positive rate} = \frac{TP}{P} \quad (4.1)$$



Obrázek 4.17: Ukázka ROC grafu, který popisuje vlastnosti pěti binárních klasifikátorů. (Převzato z [11])

Na obrázku 4.17 se nachází ukázka ROC grafu, který popisuje vlastnosti pěti různých binárních klasifikátorů. V grafu se nachází několik významných bodů, které stojí za zmínku. Bod (1,0) reprezentuje ideální klasifikátor, který se nedopouští žádných chyb. Při vývoji metod je cílem co možná nejlepší přiblížení k tomuto bodu. Dále je zde možné provést porovnání klasifikátoru A a B. Klasifikátor A je vzhledem ke klasifikátoru B poměrně konzervativní, protože kladné vzorky klasifikuje pouze v případech silné jistoty a nedopouští se tolika nesprávných detekcí v porovnání s klasifikátorem B.

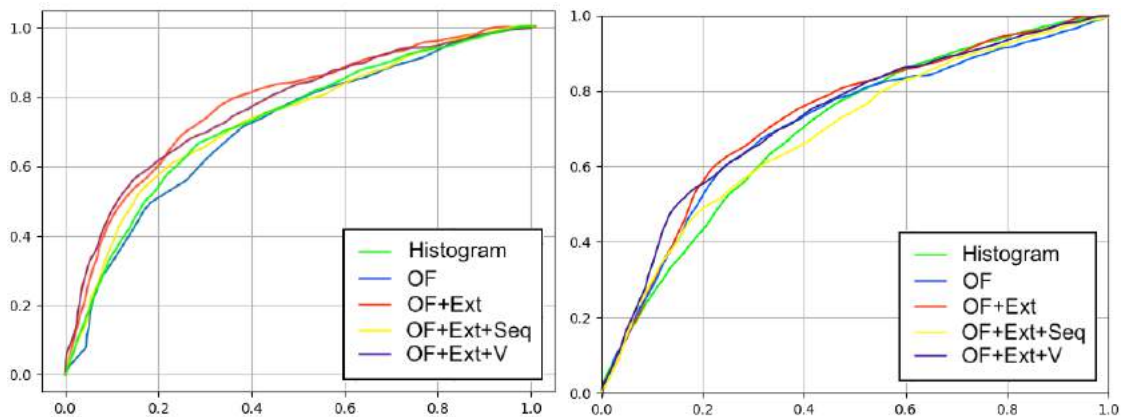
Za výchozí datovou sadu pro vyhodnocení veškerých variant detektoru anomálie jsem zvolil UMN dataset vzhledem k jednoduchosti jeho interpretace pojmu anomálie. Vzhledem



Obrázek 4.18: Výsledky vyhodnocení autoenkodéru a statistické metody na UMN datasetu a jejich porovnání s existujícími přístupy. (Převzato z [4])

k nedostupnosti trénovacích dat bez výskytu anomálie byly metody vyhodnoceny pouze v on-line režimu. Pro každou videosekvenci byl inicializován prázdný model pozadí, který byl během testování průběžně aktualizován. Scéna byla rozdělena do mřížky o velikosti 5x5 buněk. Vyhodnocení bylo provedeno z pohledu globálního výskytu anomálie a za skóre pro sestavení ROC křivky byla považována maximální hodnota z pole buněk.

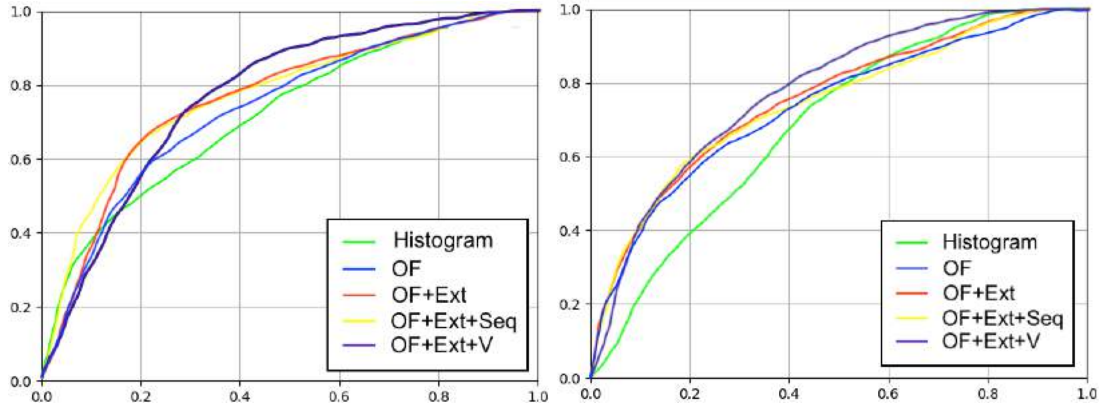
Výsledky vyhodnocení včetně porovnání s existujícími přístupy jsou uvedeny na obrázku 4.18. Srovnání variant **OF** a **OF + Ext** ukazuje na zlepšení výsledků vlivem doplnění další plně propojené vrstvy. Velmi dobré úspěšnosti rovněž dosahuje varianta využívající vizuální informace (**OF + Ext + V**). Oproti tomu výchozí metoda založená na histogramech pohybu nedokáže uspokojivě kódovat mnohdy komplexní pohybový vzor ve scéně a za metodami využívajícími konvolučního autoenkodéru zaostává.



Obrázek 4.19: Vyhodnocení navržených metod pro on-line i off-line režim na UCSD datasetu.

V dalším experimentu bylo provedeno vyhodnocení navržených metod na datové sadě UCSD. Tato datová sada již není triviální a vyskytují se zde i anomálie s projevem pouze ve vizuální doméně. Scéna byla opět rozdělena do mřížky o rozměrech 5x5 buněk a metody byly vyhodnoceny z pohledu výskytu globální anomálie. V off-line režimu byly modely pozadí pro jednotlivé buňky inicializovány na základě trénovacích sekvencí. V on-line režimu bylo

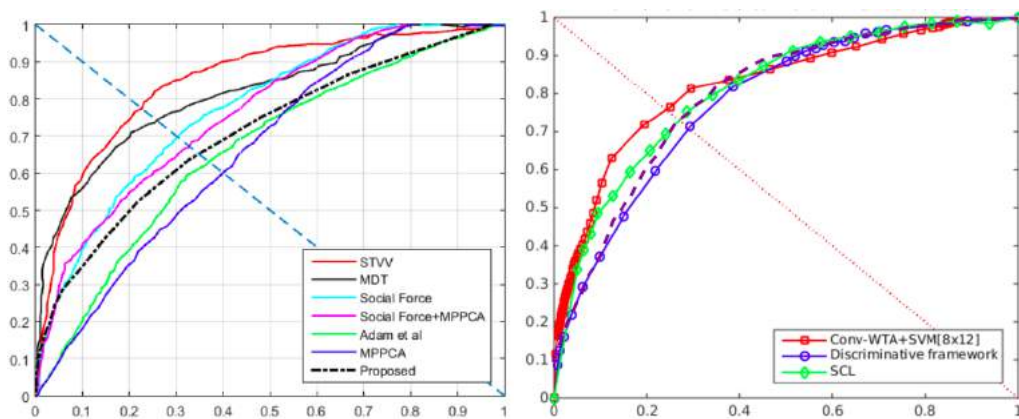
vyhodnocení spuštěno s prázdnými modely. Výsledek vyhodnocení navrhovaných metod v off-line i on-line režimu je na obrázku 4.19. Při porovnání s výsledky existujících přístupů z obrázku 4.21 je zřejmé, že navrhované metody dosahují srovnatelných výsledků v rámci obou režimů, a to i na komplexnější datové sadě.



Obrázek 4.20: Vyhodnocení navrhovaných metod pro on-line i off-line režim na Avenue datasetu. (Převzato z [37])

Zkoumané detektory anomálie byly v kontextu uživatelské úlohy navrženy pro scény velmi specifického charakteru. V rámci experimentů bylo provedeno jejich vyhodnocení i na datech odlišujících se od řešené úlohy. K tomuto účelu posloužila datová sada Avenue [22]. Výsledky vyhodnocení z obrázku 4.20 ukazují **schopnost detektorů řešit úlohu detekce anomálie obecně**. Opět bylo dosaženo srovnatelné úspěšnosti s existujícími přístupy z obrázku 4.21, a to i za on-line režimu.

Nejstabilnějších výsledků bylo v rámci veškerých experimentů dosahováno variantou kombinující optický tok společně s vizuální informací (**OF + Ext + V**). Vzhledem důležitosti vizuální informace v praktických úlohách jsem se rozhodl tuto metodu implementovat i do uživatelského nástroje. Navrhovaná metoda dosahuje v porovnání s existujícími přístupy srovnatelných výsledků a zároveň nabízí **výhodu v podobě on-line analýzy**.



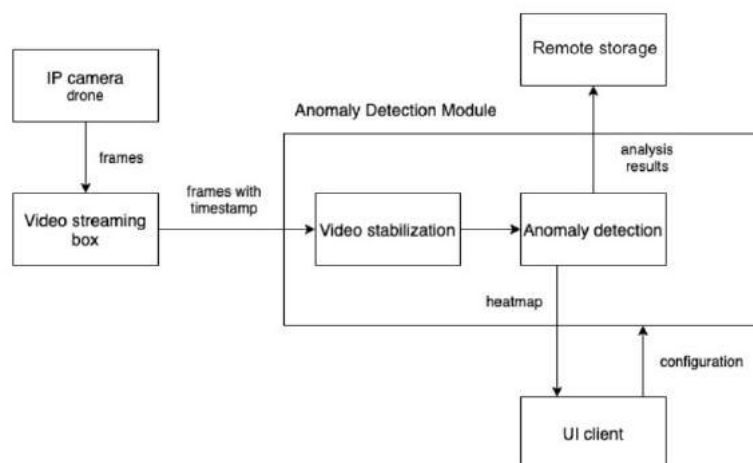
Obrázek 4.21: Vyhodnocení existujících přístupů na datových sadách UCSD a Avenue. (Převzato z [37])



## Kapitola 5

# Uživatelská aplikace

Podle detailního rozboru z kapitoly 3 lze v řešení dané úlohy identifikovat dva základní podproblémy v podobě stabilizace obrazu a detekce anomálie. Nicméně při návrhu výsledného nástroje je nutné zajistit jeho kompatibilitu s existujícím systémem pro streamování videa, vizualizační aplikací a systémem pro ukládání výstupů analýzy. Navrhovaná architektura detekčního systému je graficky znázorněna ve schématu na obrázku 5.1.



Obrázek 5.1: Navrhovaná architektura detekčního systému

Zdroj vstupní videosekvence představuje kamera umístěná na palubě dronu. Snímky z této kamery proudí směrem do video streaming boxu, který tyto snímky doplňuje o časovou značku. Systém uvažuje současný běh více výpočetních modulů na jediném zdroji videa. Z tohoto důvodu je doplnění časové značky nezbytné vzhledem k synchronizaci výsledků analýzy jednotlivých modulů.

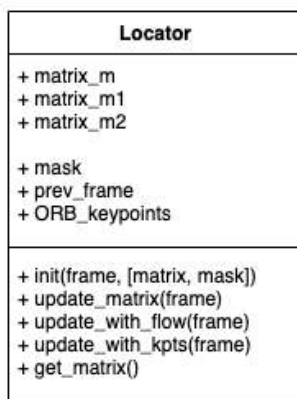
Uživatelská aplikace a výpočetní modul realizující stabilizaci videa společně s detekcí anomálie spolu komunikují podle vzoru klient-server. Výpočetní modul očekává připojení klientské aplikace. Uživatel skrze klientskou aplikaci provede volbu zdroje videa, použitých metod a dalších parametrů. Tato konfigurace je poté odeslána na stranu serveru. Po spuštění analýzy jsou ze strany serveru v pravidelných intervalech odesílány výstupy směrem ke klientské aplikaci a případně i do serveru sdružujícího výstupy z více výpočetních modulů.



Pro implementaci byl zvolen programovací jazyk Python3 [39] z důvodu rychlého prototypování a snadného provádění experimentů. Pro tento programovací jazyk jsou definována uživatelsky přívětivá aplikační rozhraní pro knihovny zabývající se zpracováním obrazu, vývojem a trénováním neuronových sítí nebo síťovou komunikací.

## 5.1 Stabilizátor videa

Problematicku stabilizace videa lze v rámci vyvíjeného nástroje považovat za samostatnou podúlohu. Z tohoto důvodu jsem se rozhodl implementovat veškerou funkcionalitu stabilizátoru pomocí třídy *Locator* vyjádřenou diagramem 5.2.



Obrázek 5.2: Objektově orientovaný návrh stabilizátoru videa.

Jediným povinným parametrem konstruktoru třídy je úvodní snímek dané videosekvence. Pomocí nepovinného parametru lze dále pomocí binární masky o velikosti úvodního snímku specifikovat stabilní oblasti v obraze. Během inicializace jsou v úvodním snímku detekovány klíčové body společně s jejich deskriptory. K jejich výpočtu bylo využito ORB detektoru z knihovny OpenCV [6].

Nejdůležitějšími atributy třídy *Locator* je trojice transformačních matic. Transformační matice *m1* definuje transformaci mezi úvodním snímkem a souřadnicovým systémem mapového podkladu. Tato matice představuje uživatelský vstup a dané třídě může být předána pomocí nepovinného argumentu konstruktoru.

Transformace mezi původním a aktuálním snímkem videa je vyjádřena pomocí matice *m2*. Tato matice je nejprve inicializována jako identita a s každým dalším snímkem je provedena její aktualizace. Zbývající transformační matice vyjadřuje transformaci mezi dvojicí posledních zpracovaných snímků.

Stabilizace videa je realizována pomocí metody `update_matrix()`, jejíž parametrem je aktuální snímek videosekvence. Stabilizátor nejprve předpokládá výskyt pouze drobných záchvěvů. Pomocí Harrisova detektoru je nejprve provedena detekce vhodných klíčových bodů na předchozím snímku. Tyto body jsou pomocí Lucas-Kanade metody sledovány do aktuálního snímku. Na základě odpovídajících dvojic bodů je poté vypočtena homografie mezi těmito snímky. Při implementaci bylo opět využito knihovních funkcí OpenCV.

Pro stabilizaci drobných záchvěvů je definován minimální počet klíčových bodů, na základě kterých je výpočet homografie prováděn. V případě nedostatečného počtu bodů není transformace považována za relevantní a výpočet selhává. Druhým důvodem možného

selhání výpočtu je přílišná míra záchvěvu. V těchto případech jsou v aktuálním snímku videa detekovány klíčové body společně s ORB deskriptory a pomocí Brute-force matcheru knihovny OpenCV jsou k nim nalezeny odpovídající body z původního snímku.

I v tomto případě je definovaný minimální počet bodů, na základě kterého musí být homografie vypočtena. V případě nesplnění této podmínky se transformaci nepodařilo nalézt a další pokus o dohledání transformace s původní scénou bude proveden až s dalším snímkem.

## 5.2 Detektor anomálie

Detekce anomálie je podobně jako stabilizace obrazu samostatnou úlohou, kterou je vhodné implementovat pomocí objektově orientovaného přístupu. Vzhledem k implementaci dvou zcela odlišných detektorů bylo nutné provést specifikaci jejich společného rozhraní. Povinným argumentem konstruktoru třídy `Model` je počet buněk tvořících model pozadí. Pomocí nepovinných argumentů mohou být dále definovány specifické parametry jednotlivých přístupů.

Výpočet teplotní mapy na základě aktuálního optického toku je realizován metodou `update_model()`. Třída `Model` je zodpovědná za rozdělení optického toku mezi jednotlivé buňky a za výpočet příznakového vektoru pro každou jeho sekci. V případě statistické metody představuje příznakový vektor 2D histogram rozložení rychlostí a směrů pohybu, v druhém případě je příznakem výstup neuronové sítě.

Atributem třídy `Model` je seznam jednotlivých buněk mřížky implementovaných třídou `ModelCell`. Pomocí metody `update(vector)` je aktualizován model pozadí dané buňky odpovídajícím příznakovým vektorem. Metoda `get_anomaly_level(vector)` slouží k samotnému výpočtu úrovně anomálie pro daný příznak. Po specifikaci společné struktury zde budou dále uvedeny implementační specifika jednotlivých metod.

Společným vstupem obou metod je hustý optický tok vypočtený pomocí Farnebackovy metody implementované knihovnou OpenCV [6]. Tato metoda splňuje podmínku možnosti zpracování videa v reálném čase a zároveň dosahuje dostatečných výsledků v kvalitě výpočtu. Knihovna Numpy [16] nabízí široké množství výpočetně efektivních funkcí pro výpočet 2D histogramu i realizaci maticových operací nutných k implementaci základní statistické metody.

Neuronová síť metody založené na extrakci příznakového vektoru byla implementována s využitím frameworku Keras [15]. Jedná se o velmi často používanou knihovnu poskytující aplikační rozhraní jazyka Python3 [39] pro experimentování s umělou inteligencí. Tato knihovna dále umožňuje akceleraci výpočtu pomocí grafické karty, což je přínosné zejména pro realtime aplikace.

Příznakové vektory extrahované pomocí neuronové sítě jsou modelovány pomocí směsi Gaussovských rozložení [31]. Nebyla nalezena žádná vhodná knihovna implementující učení tohoto modelu po jednotlivých dávkách v reálném čase, a proto jsem se musel uchýlit k implementaci vlastního řešení.

Ke každé Gaussově křivce z GMM modelu je uchováván celkový počet vzorků a jejich lineární a kvadratický součet. Na základě těchto statistik lze při libovolné fázi učení po dávkách určit aktuální střední hodnotu a varianci v jednotlivých dimenzích příznaku. K výpočtu hodnoty funkce hustoty rozdělení pravděpodobnosti bylo využito knihovny Scipy [40] a veškeré maticové operace byly z důvodu efektivity opět implementovány pomocí knihovny Numpy [16].

Parametr	Popis
Zdroj videa	Jedná se o textový řetězec specifikující vybranou IP kameru nebo název souboru.
Stabilní oblasti	Binární maska o rozlišení vstupního videa. Definuje stabilní oblasti ve videu na počátečním snímku analýzy.
Transformační matice	Definuje transformaci počátečního snímku videa do modelového prostoru.
Parametry modelu	Načtení/ukládání modelu pozadí, pozastavení aktualizace modelu.
Použitá metoda	Textový řetězec specifikující použitou metodu detekce anomálie.

Tabulka 5.1: Popis parametrů zasílaných klientskou aplikací výpočetnímu serveru.

### 5.3 Výpočetní modul a klientská aplikace

V předchozích sekcích byl proveden důkladný rozbor problematiky stabilizace videa a detekce anomálie z implementačního hlediska. Nyní je nutné rozhodnout o vhodné architektuře uživatelského nástroje z pohledu praktického nasazení. Z popisu uživatelských úloh v kapitole 3.1 je zřejmé nasazení nástroje v terénu. V případě výskytu anomálie v rámci snímaného prostoru je z pohledu zasahující policejní hlídky výhodné, aby zde byla možnost sledovat vývoj situace na přenosném zařízení. Sledováno může být i více scén nezávisle na sobě, což opět komplikuje situaci z pohledu výpočetních nároků. Z tohoto důvodu bylo provedeno rozdělení výpočetní a vizualizační části nástroje.

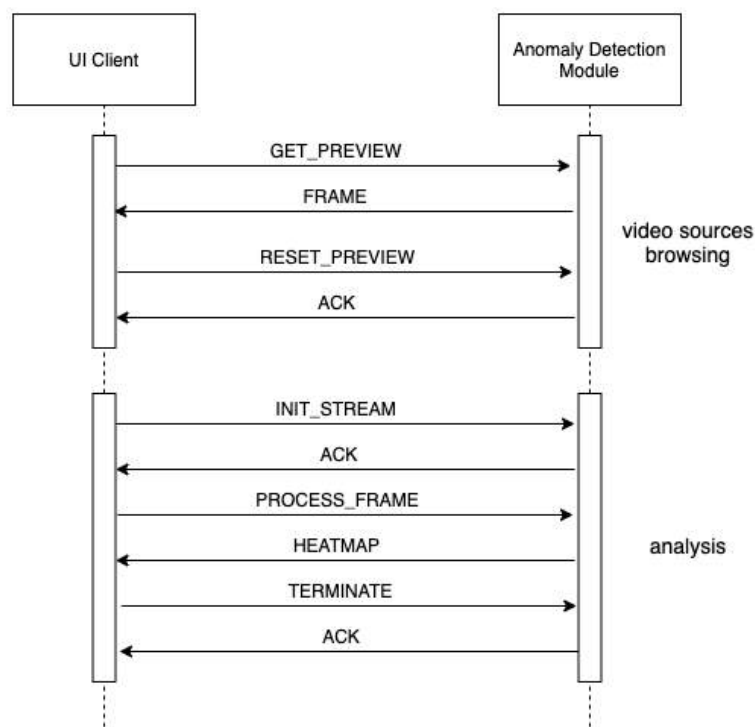
Komunikace mezi klientskou aplikací a výpočetním modulem je založena na výměně zpráv skrze síťový socket. Pro implementaci síťové komunikace byla využita stejnojmenná knihovna jazyka Python3 [38]. Zpráva je definována jako dvojice číselného kódu a serializovaných binárních dat. Číselný kód definuje, jakým způsobem mají být zasílaná data na straně serveru zpracována. Serializace datových struktur programovacího jazyka Python3 do binární podoby je zajištěna pomocí knihovny Pickle [38].

Číselné kódy komunikačního protokolu lze rozdělit do dvou skupin ilustrovaných diagramem 5.3. První skupina umožňuje procházení dostupných videostreamů nebo záznamů. Důvodem je možnost spuštění analýzy od libovolného snímku v rámci videosekvence. Díky tomu lze zajistit, aby nebyla analýza spuštěna hned od počátku letu dronu a během jeho manévrování.

Druhá skupina příkazů umožňuje zasílání uživatelských konfigurací ze strany klienta, řízení provádění analýzy a na závěr i ukončení komunikace. Uživatelské parametry jsou na server odesílány v podobě serializované datové struktury `dictionary` [39]. Shrnutí nejzákladnějších položek je uvedeno v tabulce 5.1.

Úlohou výpočetního modulu je zajištění výpočetních prostředků pro úlohu stabilizace videa a detekce anomálie. Pro výpočetní modul není potřebné žádné grafické uživatelské rozhraní a předpokládá se jeho spuštění na vzdáleném serveru. Výpočetní modul je specifikován IP adresou a číslem portu, na kterém očekává klientské dotazy.

Pomocí metody `init_server()` je provedena inicializace socketu, na kterém je očekáváno připojení klientských aplikací. Metodou `start()` je zahájeno čekání na příchozí klienty. Parametrem této funkce je instance obecné třídy `Method` definující rozhraní pro výpočetní modul. Díky tomu není server omezený pouze na jediný typ výpočetního modulu. Každý

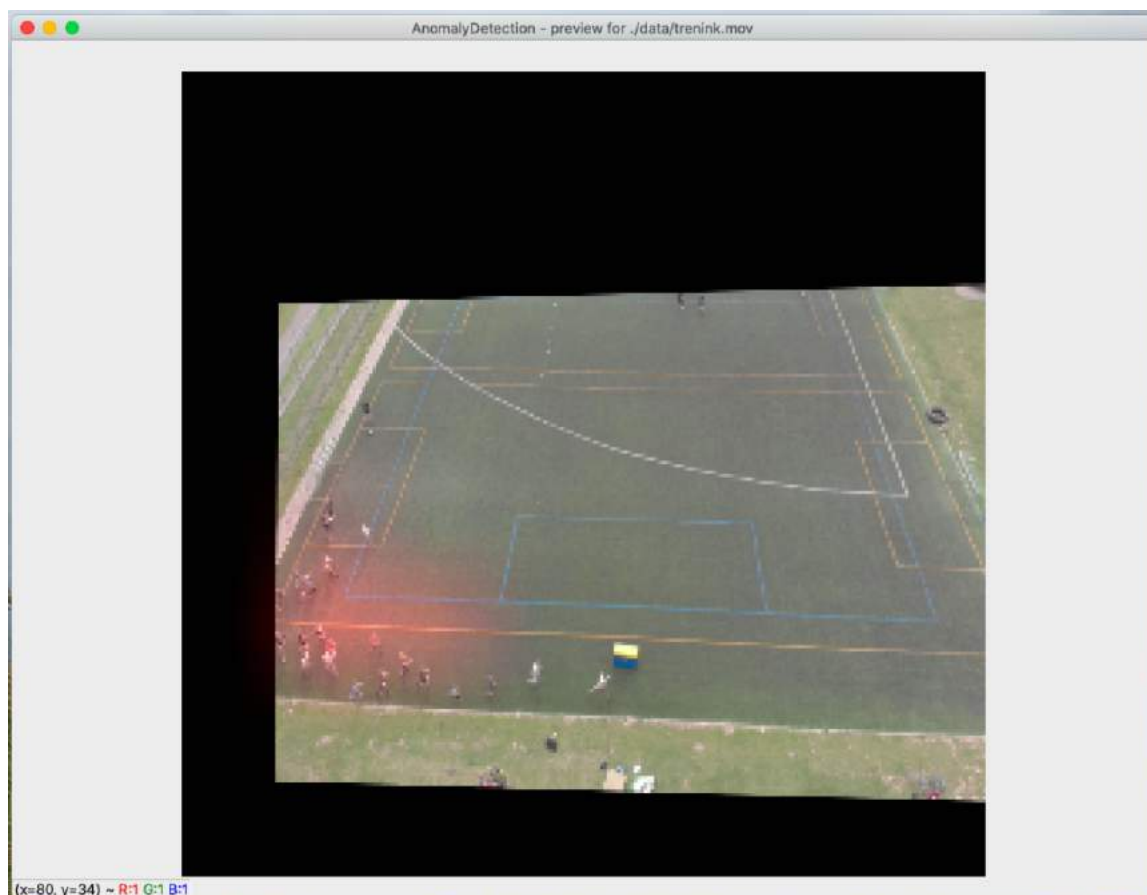


Obrázek 5.3: Schématické znázornění komunikačního protokolu mezi výpočetním modulem a uživatelským klientem.

výpočetní modul musí implementovat funkci `process()`. Pomocí této metody přijímá výpočetní modul zprávy z klientské aplikace a reaguje na ně korektní návratovou hodnotou.

Klientská aplikace vyvíjená v rámci této práce představuje terminálovou aplikaci a slouží zejména pro účely vývoje. Vzhledem k jasně definovanému komunikačnímu rozhraní a oddělení výpočetní a vizualizační části je zde připraven prostor pro její nahrazení uživatelsky přívětivějším nástrojem.

Po spuštění aplikace je nejprve otevřeno okno, jehož cílem je nalezení transformace mezi mapovým podkladem a počátečním snímkem videosekvence. Úkolem uživatele je označení dvojic odpovídajících bodů mapového podkladu a snímku videa. Skrze druhé dialogové okno je uživatelem zadávána binární maska stabilních oblastí. Po tomto kroku již následuje spuštění vizualizace výstupu analýzy, jejíž ukázka je na obrázku 5.4. Videodata jsou zde transformována do prostoru scény. K signalizaci zvýšené úrovně anomálie bylo využito červené barvy. Veškeré GUI prvky byly realizovány pomocí knihovny OpenCV [6].



Obrázek 5.4: Ukázka vizualizace výstupu analýzy pomocí grafického uživatelského rozhraní klientské aplikace.

## Kapitola 6

# Závěr

Cílem této práce byl návrh a implementace nástroje, který umožňuje automatizovanou analýzu chování davu na videodatech z dronu a napomáhá lokalizovat případnou anomálii na mapovém podkladu. Nejprve byl proveden průzkum existujících přístupů pro detekci anomálie. Jejich bližší prozkoumání poté vedlo k identifikaci metod využitelných pro tuto úlohu. Vzhledem k specifikům videodat z dronu musely být prozkoumány i vhodné způsoby jejich stabilizace a registrace do mapového podkladu.

Na základě teoretických poznatků proběhl návrh metody stabilizace videa specializované na zvláštnosti videodat z dronu. Tato metoda zajišťuje stabilizaci záchrův drobného charakteru, ale je připravena i na výrazné záchrůvy projevující se kompletní ztrátou původní scény. Vyhodnocení této metody bylo provedeno na vlastních datech z reálného prostředí a byla prokázána její spolehlivost. Pořízený dataset se skládá z 9 videosekvencí opatřených manuálně pořízenou anotací.

Za hlavní přínos práce považuji návrh dvou nových detektorů anomálie. Jeden z nich využívá základních přístupů k analýze optického toku a posloužil zejména jako výchozí varianta při vývoji kompletního detekčního systému. Druhý detektor využívá neuronové sítě s architekturou autoenkodéru pro extrakci příznakového vektoru. Obvyklé chování na základě příznaků poté modeluje pomocí vlastní implementace směsi Gaussovských rozložení, což zajišťuje možnost on-line zpracování.

Nad konfigurací detektorů bylo provedeno mnoho experimentů a jejich funkce byla na závěr vyhodnocena na třech nejznámějších datových sadách zaměřených na detekci anomálie. Jejich úspěšnost je srovnatelná s nejlepšími přístupy současnosti, ale zároveň nabízí i výhodu v podobě možnosti zpracování dat v on-line režimu. Dále se ukázalo, že jejich použití není omezeno jenom na davové scény snímání z paluby dronu, ale řeší úlohu detekce anomálie obecně. Funkčnost nástroje byla na závěr integrována do demonstrační terminálové aplikace.

# Literatura

- [1] ALI, S. a SHAH, M. A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In: IEEE. *2007 IEEE Conference on Computer Vision and Pattern Recognition*. 2007, s. 1–6.
- [2] BAKER, S. a MATTHEWS, I. Lucas-kanade 20 years on: A unifying framework. *International journal of computer vision*. Springer. 2004, sv. 56, č. 3, s. 221–255.
- [3] BANK, D., KOENIGSTEIN, N. a GIRYES, R. Autoencoders. *ArXiv preprint arXiv:2003.05991*. 2020.
- [4] BANSOD, S. a NANDEDKAR, A. Detection and Localization of Anomalies from Videos based on Optical flow Magnitude and Direction. In: Atlantis Press. *International Conference on Communication and Signal Processing 2016 (ICCASP 2016)*. 2016, s. 454–462.
- [5] BARLA, A., ODONE, F. a VERRI, A. Histogram intersection kernel for image classification. In: IEEE. *Proceedings 2003 international conference on image processing (Cat. No. 03CH37429)*. 2003, sv. 3, s. III–513.
- [6] BRADSKI, G. a KAEHLER, A. *Learning OpenCV: Computer vision with the OpenCV library*. "O'Reilly Media, Inc.", 2008.
- [7] CHAN, A. a VASCONCELOS, N. Ucsd pedestrian dataset. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*. 2008, sv. 30, č. 5, s. 909–926.
- [8] DERPANIS, K. G. The harris corner detector. *York University*. Citeseer. 2004, sv. 2.
- [9] DOSOVITSKIY, A., FISCHER, P., ILG, E., HAUSSE, P., HAZIRBAS, C. et al. FlowNet: Learning optical flow with convolutional networks. In: *Proceedings of the IEEE international conference on computer vision*. 2015, s. 2758–2766.
- [10] FARNEBÄCK, G. Two-frame motion estimation based on polynomial expansion. In: Springer. *Scandinavian conference on Image analysis*. 2003, s. 363–370.
- [11] FAWCETT, T. An introduction to ROC analysis. *Pattern recognition letters*. Elsevier. 2006, sv. 27, č. 8, s. 861–874.
- [12] FISCHLER, M. A. a BOLLES, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*. ACM New York, NY, USA. 1981, sv. 24, č. 6, s. 381–395.



- [13] GOODFELLOW, I., BENGIO, Y. a COURVILLE, A. *Deep Learning*. MIT Press, 2016.  
<http://www.deeplearningbook.org>.
- [14] GUILLUY, W., OUDRE, L. a BEGHDAI, A. Video stabilization: challenges, methods and evaluation.
- [15] GULLI, A. a PAL, S. *Deep learning with Keras*. Packt Publishing Ltd, 2017.
- [16] HARRIS, C. R., MILLMAN, K. J., WALT, S. J. van der, GOMMERS, R., VIRTANEN, P. et al. Array programming with NumPy. *Nature*. Springer Science and Business Media LLC. září 2020, sv. 585, č. 7825, s. 357–362. DOI: 10.1038/s41586-020-2649-2. Dostupné z: <https://doi.org/10.1038/s41586-020-2649-2>.
- [17] HOCHREITER, S. a SCHMIDHUBER, J. Long short-term memory. *Neural computation*. MIT Press. 1997, sv. 9, č. 8, s. 1735–1780.
- [18] HOSSAIN, M. A. a HUH, E.-N. A Novel Performance Metric for the Video Stabilization Method. In: Springer. *International Conference on Ubiquitous Information Management and Communication*. 2019, s. 486–496.
- [19] KIRAN, B. R., THOMAS, D. M. a PARAKKAL, R. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *Journal of Imaging*. Multidisciplinary Digital Publishing Institute. 2018, sv. 4, č. 2, s. 36.
- [20] LI, T., CHANG, H., WANG, M., NI, B., HONG, R. et al. Crowded scene analysis: A survey. *IEEE transactions on circuits and systems for video technology*. IEEE. 2014, sv. 25, č. 3, s. 367–386.
- [21] LOWE, D. G. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*. Springer. 2004, sv. 60, č. 2, s. 91–110.
- [22] LU, C., SHI, J. a JIA, J. Abnormal event detection at 150 fps in matlab. In: *Proceedings of the IEEE international conference on computer vision*. 2013, s. 2720–2727.
- [23] MADEMLIS, I., MYGDALIS, V., NIKOLAIDIS, N., MONTAGNUOLO, M., NEGRO, F. et al. High-level multiple-UAV cinematography tools for covering outdoor events. *IEEE Transactions on Broadcasting*. IEEE. 2019, sv. 65, č. 3, s. 627–635.
- [24] MADEMLIS, I., NIKOLAIDIS, N., TEFAS, A., PITAS, I., WAGNER, T. et al. Autonomous UAV Cinematography: A Tutorial and a Formalized Shot-Type Taxonomy. *ACM Computing Surveys (CSUR)*. ACM New York, NY, USA. 2019, sv. 52, č. 5, s. 1–33.
- [25] MAHADEVAN, V., LI, W., BHALODIA, V. a VASCONCELOS, N. Anomaly detection in crowded scenes. In: IEEE. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2010, s. 1975–1981.
- [26] MEDEL, J. R. a SAVAKIS, A. Anomaly detection in video using predictive convolutional long short-term memory networks. *ArXiv preprint arXiv:1612.00390*. 2016.
- [27] MUJA, M. a LOWE, D. G. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*. 2009, sv. 2, 331–340, s. 2.

- [28] RAGHAVENDRA, R., BUE, A. a CRISTANI, M. *Unusual crowd activity dataset of University of Minnesota*. 2006.
- [29] RAVANBAKHS, M., NABI, M., MOUSAVI, H., SANGINETO, E. a SEBE, N. Plug-and-play cnn for crowd motion analysis: An application in abnormal event detection. In: IEEE. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2018, s. 1689–1698.
- [30] RAVANBAKHS, M., NABI, M., SANGINETO, E., MARCENARO, L., REGAZZONI, C. et al. Abnormal event detection in videos using generative adversarial nets. In: IEEE. *2017 IEEE International Conference on Image Processing (ICIP)*. 2017, s. 1577–1581.
- [31] REYNOLDS, D. A. Gaussian Mixture Models. *Encyclopedia of biometrics*. Berlin, Springer. 2009, sv. 741, s. 659–663.
- [32] ROJAS, R. Lucas-kanade in a nutshell. *Freie Universit at Berlinn, Dept. of Computer Science, Tech. Rep*. 2010.
- [33] RUBLEE, E., RABAUD, V., KONOLIGE, K. a BRADSKI, G. ORB: An efficient alternative to SIFT or SURF. In: Ieee. *2011 International conference on computer vision*. 2011, s. 2564–2571.
- [34] SONKA, M., HLAVAC, V. a BOYLE, R. *Image processing, analysis, and machine vision*. Nelson Education, 2014.
- [35] SRIVASTAVA, N., MANSIMOV, E. a SALAKHUDINOV, R. Unsupervised learning of video representations using lstms. In: *International conference on machine learning*. 2015, s. 843–852.
- [36] SVOZIL, D., KVASNICKA, V. a POSPICHAL, J. Introduction to multi-layer feed-forward neural networks. *Chemometrics and intelligent laboratory systems*. Elsevier. 1997, sv. 39, č. 1, s. 43–62.
- [37] TRAN, H. T. a HOGG, D. Anomaly detection using a convolutional winner-take-all autoencoder. In: British Machine Vision Association. *Proceedings of the British Machine Vision Conference 2017*. 2017.
- [38] VAN ROSSUM, G. *The Python Library Reference, release 3.8.2*. Python Software Foundation, 2020.
- [39] VAN ROSSUM, G. a DRAKE, F. L. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. ISBN 1441412697.
- [40] VIRTANEN, P., GOMMERS, R., OLIPHANT, T. E., HABERLAND, M., REDDY, T. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*. 2020, sv. 17, s. 261–272. DOI: 10.1038/s41592-019-0686-2.
- [41] XU, D., RICCI, E., YAN, Y., SONG, J. a SEBE, N. Learning deep representations of appearance and motion for anomalous event detection. *ArXiv preprint arXiv:1510.01553*. 2015.

- [42] ZHANG, T., RAMAKRISHNAN, R. a LIVNY, M. BIRCH: an efficient data clustering method for very large databases. *ACM sigmod record*. ACM New York, NY, USA. 1996, sv. 25, č. 2, s. 103–114.
- [43] ZHOU, B., WANG, X. a TANG, X. Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents. In: IEEE. *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2012, s. 2871–2878.

## Příloha A

# Obsah přiloženého paměťového média

- `src` - Zdrojové soubory terminálové aplikace.
- `autoencoder` - Trénovací skripty a natrénované modely konvolučního autoenkodéru.
- `thesis` - Digitální verze diplomové práce.
- `demos` - Videoukázky funkce nástroje.
- `data` - Videosekvence sloužící pro účely demonstrace funkce nástroje.
- `dataset_STAB` - Soubory vlastního datasetu.
- `dataset_AVENUE` - Soubory datové sady Avenue.
- `dataset_UCSD` - Soubory datové sady UCSD.
- `dataset_UMN` - Soubory datové sady UMN.